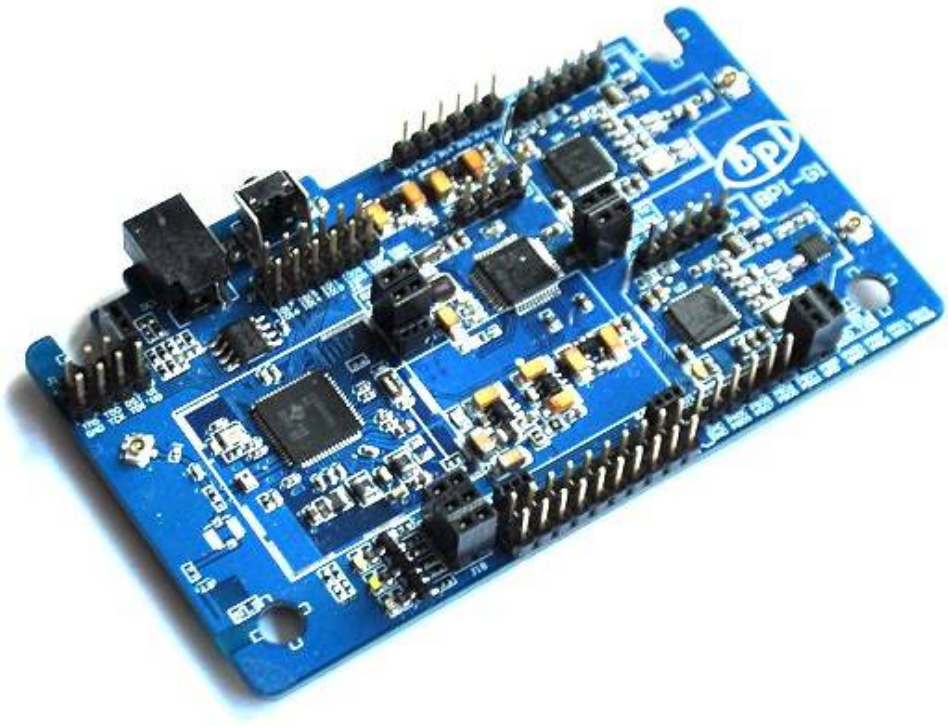
BPI G1 User Manual

V1.0



Catalog

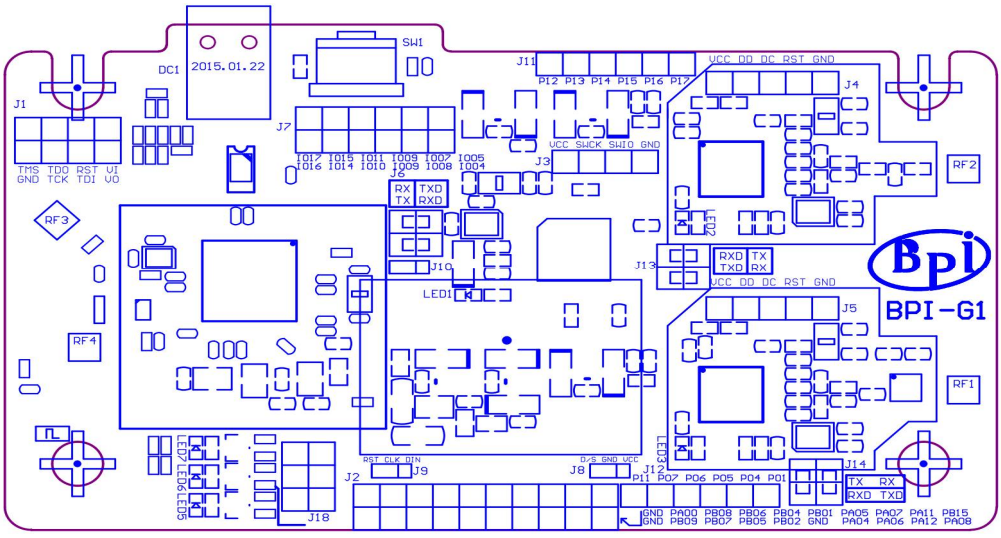
BPI G1 Instructions....1
Catalog....2
1 Introduction....4
1.1 BPI G1....4
1.2 Main features....7
1.2 Which IDE development can use?....8
1.3 Which emulator debugging?....8
2 Hardware Description....8
2.1 Some of the features TI CC3200....8
2.2 Some of the features TI CC2530....10
2.3 Some of the features TI CC2540/1....10
2.4 Some of the features STM32F103CB....11
3 Peripheral Interface....12
3.1 TI CC3200 part Peripheral Interface....12
3.1.1 4-wire SDIO Interface....12
3.1.2 Uart Interface....13
3.1.3 SPI Interface....13
3.1.4 I2C Interface....14
3.1.5 PWM & Timer Capture channel....14
3.16 ADC channel....15
3.2 TI CC2530 part Peripheral Interface....15
3.2.1 Uart Interface....15
3.2.2 SPI Interface....16
3.2.3 PWM & Timer Capture channel....16
3.2.4 ADC channel....17
3.3 TI CC2540/1 part Peripheral Interface....17
3.3.1 Uart Interface....17
3.3.2 SPI Interface....18
3.3.3 ADC channel....18
3.3.4 PWM & Timer Capture channel....19
3.4 STM32F103CB part Peripheral Interface....19
3.4.1 Uart Interface....19
3.4.2 SPI Interface....20
3.4.3 I2C Interface....20
3.4.4 CAN Interface....21
3.4.5 USB2.0 Interface....21
3.4.6 PWM & Timer Capture channel....22
3.4.7 ADC channel....22
4 Global Power and Ground....23

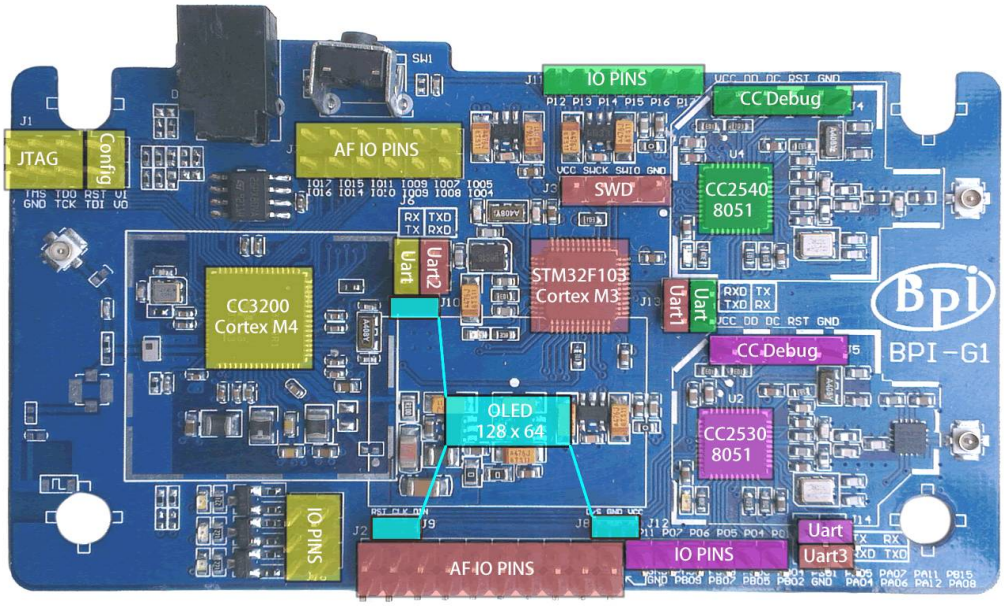
5 Module interconnection 23
5.1 CC3200 connection with F103 23
5.2 CC2540 connection with F103 24
5.3 CC2530 connection with F103 24
6 Status LEDs for each module 25
7 TI CC3200 mode selection jumper 25
7.1 Programming mode 25
7.2 Run mode 26
8 Button 27
9 How to Quick Start? 27
9.1 Quick start of TI CC3200 27
9.1.1 Download and install the software 27
9.1.2 Configuration G1 28
9.1.3 G1 connection BPI OPENDEBUGGER 29
9.1.4 Check whether the connected BPI OPENDEBBUG 29
9.1.5 Cover IAR debugger interface file 30
9.1.6 Recompile SimpleLink library files 30
9.1.7 Compile, download and debug routines WLAN Station 31
9.2 Quick start of TI CC2540/1 33
9.2.1 Download and install the software 34
9.2.2 G1 connection BPI OPENDEBUGGER 34
9.2.3 BPI OPENDEBBUG check whether the software and hardware are connected 34
9.2.4 Compile, download and debug routines SimpleBLEPeripheral 35
9.3 Quick start of TI CC2530 38
9.3.1 G1 connection BPI OPENDEBUGGER 39
9.3.2 BPI OPENDEBBUG check whether the software and hardware are connected 40
9.3.3 Modifications to adapt the code for G1 40
9.3.4 Compile, download and debug routines SampleApp 42
9.4 Quick Start STM32F103CB 46
9.4.1 G1 connection BPI OPENDEBUGGER 47
9.4.2 BPI OPENDEBBUG check whether the software and hardware are connected 47
9.4.3 Compile, download and debug routines SimpleLed 48
10 Explanation 50

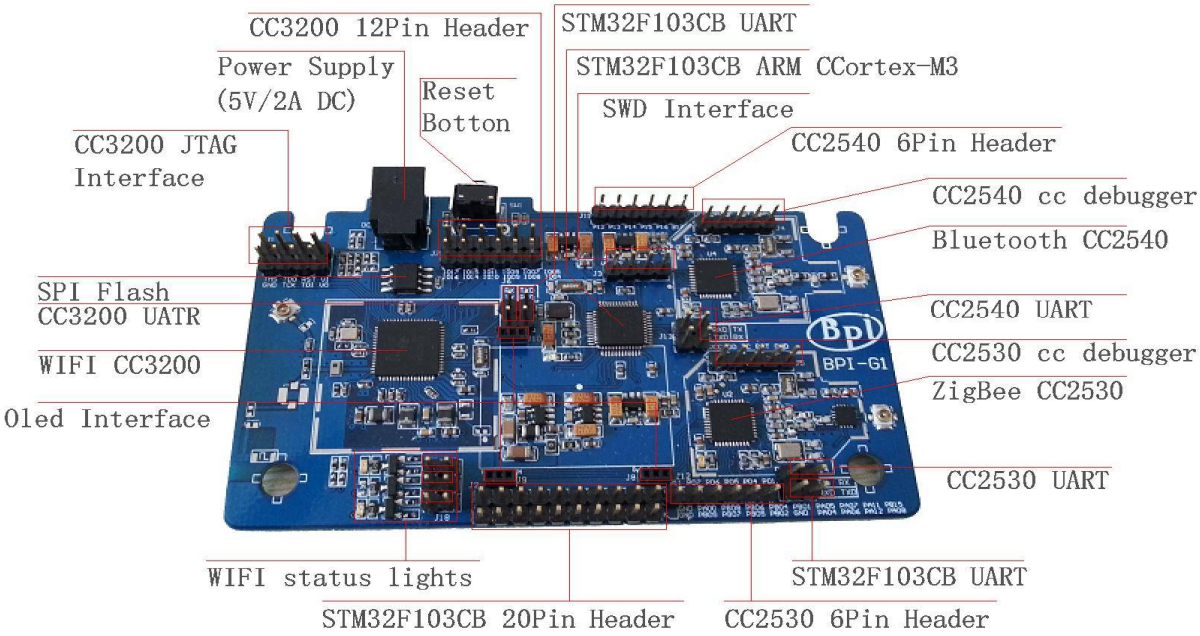
1 Introduction

1.1 BPI-G1

BPI-G1 is an integrated IEEE802.11 b / g / n (WIFI wireless network), IEEE802.15.4 (Zigbee), IEEE 802.11-2007 Standard (Bluetooth Low Energy4.0) development board. It is very easy to put these wireless protocol seamlessly together, you can exchange any different transport protocols, and three kinds of wireless protocols are integrated by three single-chip SOC, you can easily create your own Internet of things (IoT). WIFI part of which is the use of TI's CC3200, which is a high-performance ARM® Cortex®-M4 wireless SOC, internally integrated TCP / IP protocol stack, simply use the BSD Socket can connect to the Internet. The Zigbee is used TI CC2530, which integrates wireless capabilities and enhanced 8051 core SOC, TI after years of improvement, it is already quite mature and stable, TI's Z-stack has achieved Zigbee 2007 / Pro, you can use the 16's short address, you can use the 64-bit long address communication, face large local interconnect systems, advanced security encryption and mesh network structure can easily cope. Bluetooth 4.0 (BLE) using TI CC2540 / 1, an integrated BLE stack and enhanced 8051 core, low-power wireless SOC, at present, most mobile phones have support for Bluetooth 4.0, both as a wearable device, or mobile interactive Accessories, CC2540 can be easily completed. Meanwhile BPI G1 also incorporates a high-performance STM32 ARM® Cortex®-M3 microcontroller, which will help you deal with time-consuming data or transit, the three wireless SOC coordinated. Therefore, the use of BPI G1 can help you quickly achieve a variety of things DIY design.




IO PINS
CC Debug
JTAG
Config
AF IO PINS
SWD
CC2540
8051
Uart
Uart2
STM32F103
Cortex M3
Uart1
Uart
CC3200
Cortex M4
CC Debug
BPI-G1
OLED
128 x 64
CC2530
8051
IO PINS
Uart
Uart3
AF IO PINS
IO PINS


CC3200 12Pin Header
STM32F103CB UART
Power Supply
(5V/2A DC)
Reset
Botton
STM32F103CB ARM CCortex-M3
SWD Interface
CC3200 JTAG
Interface
CC2540 6Pin Header
CC2540 cc debugger
Bluetooth CC2540
SPI Flash
CC3200 UATR
CC2540 UART
WIFI CC3200
CC2530 cc debugger
ZigBee CC2530
Oled Interface
CC2530 UART
WIFI status lights
STM32F103CB UART
STM32F103CB 20Pin Header
CC2530 6Pin Header

Hardware Specifications

Items	Hardware Specifications
CPU	STM32F103CB ARM Cortex™-M3 32-bit RISC core
WIFI	CC3200 device is an integrated high-performance ARM Cortex-M4 MCU and with a Wi-Fi network processor subsystem (This subsystem includes 802.11 b / g / n radio, baseband, network protocol stack and a powerful encryption engine MAC, supports 256-bit encryption in order to achieve a fast, secure Internet connection). This device contains a variety of peripherals, I2S, SD / MMC, UART, SPI, I2C and four-channel analog to digital converter (ADC).
Bluetooth	CC2540 device is a low cost, low power, true system-on-chip (SoC) for Bluetooth low energy applications. It enables to build a strong BLE master or slave node with very low total cost BOM. The CC2540 combines the excellent RF transceiver, the industry-standard enhanced 8051 MCU, in-system programmable flash memory, 8 KB RAM and many other powerful auxiliary functions and peripherals.
Zigbee	CC2530 device is a true system-on-chip (SoC) for IEEE802.15.4, Zigbee RF4CE applications and solutions. It enables the establishment of a strong network of nodes and very low total cost BOM. CC2530 combines leading RF transceiver, the industry-standard enhanced 8051 MCU, in-system programmable flash memory, 8 KB RAM

	outstanding performance and many other powerful features.
Power	5V DC
Buttons	Reset button
LED	WiFi Bluetooth Zigbee
Monitor	OLED（128x64）
OS	Free-RTOS、Ti-OS、Coustom-OS
Product size	95mm X 56mm
Weight	19g

What can I do with BPI-G1 ?

1.Smart Home.。

2.Intelligent Gateway。

3.DIY electronic control。

1.2 Main features

• Multi-module integration, development board does not occupy a small space
• Multi-protocol integration, to meet the Internet of things needs of various wireless connections
• Rich interface, no matter which one you want to achieve functional modules can easily match
• Abundant development resources, open source sharing of data and TI, ST and other major manufacturers improve the development of information
• IOS and Android devices open software interface helps you quickly connect a handheld control device
• Zigbee wireless integrated power amplifier section, an open area can reach as far as 3 km of intercellular communication

1.2 Which IDE development can use?

TI CC3200：

Code Composer Studio (CCS)
IAR EWARM
GCC

TI CC2530/CC2540：

IAR EW8051
Instant Contiki

STM32F103CB：

IAR EWARM
GCC

1.3 Which emulator debugging?

TI C3200：

BPI OpenDebugger

TI CC2530/CC2540：

BPI OpenDebugger
TI CC Debugger
TI SmartRF04EB
TI SmartRF05EB

STM32F103CB：

BPI OpenDebugger
JLink V8
STLink V2

2 Hardware Description

2.1 Some of the features TI CC3200

- ARM Cortex-M4 core, 80MHz operating frequency
- Embedded memory RAM (up to 256KB)
- External serial flash boot loader, and the ROM peripheral drivers
- 32-channel direct memory access (DMA)

- For advanced fast security hardware encryption engine, including
  AES，DES and 3DES
  SHA2 and MD5
  Cyclic Redundancy Check (CRC) checksum
- 1 SD / MMC interface
- 2 Universal Asynchronous Receiver Transmitter (UART)
- 1 serial peripheral interface (SPI)
- 1 inter-integrated circuit (I2C)
- 4 general-purpose timers, support for 16-bit pulse width modulation (PWM) mode
- 1 watchdog timer
- 4-channel 12-bit ADC (ADC)
- Up to 12 independently programmable, reusable general purpose input output (GPIO) pins
- Built-in TCP / IP stack
  Industry standard BSD socket application programming interface (API)
  8 simultaneous TCP sockets or UCP
  2 slots while TLS and SSL
- Strong encryption engine for the 256 for AES TLS and SSL encrypted connections fast, secure Wi-Fi and internet connection
- Base stations, access points (AP) and Wi-Fi Direct Mode
- WPA2 Personal and Enterprise Security
- For independent and fast Wi-Fi connection SimpleLink Connection Manager
- SmartConfig technology, AP mode and WPS2, these techniques used to implement a simple and flexible Wi-Fi hotspot
- Tx power
  18.0 dBm @ 1 DSSS
  14.5 dBm @ 54 OFDM
- RX Sensitivity
  -95.7 dBm @ 1 DSSS
  -74.0 dBm @ 54 OFDM
- Advanced low power mode
  Support real-time clock (RTC) Sleep: 4 μA
  Low power deep sleep (LPDS): 120 μA
  RX flow (MCU activation): 59 mA @ 54 orthogonal frequency division multiplexing (OFDM),
  TX flow (MCU activation): 229 mA @ 54OFDM, maximum power
  Idle connections (in LPDS in MCU): 695 μA @ DTIM = 1
- Clock Source
  40.0MHz crystal with internal oscillator
  RTC 32.768kHz crystal or external clock

2.2 Some of the features TI CC2530

- Excellent performance and low power 8051 microcontroller core with code prefetch feature
- 256K system programmable flash memory
- 8KB RAM, with a power supply in a variety of data retention
- Hardware debugging support
- The powerful 5-channel DMA
- Integrated high-performance operational amplifiers and ultra low-power comparator
- IEEE 802.15.4 MAC timer, general-purpose timers (one 16 bit, two 8 bit)
- IR generating circuit
- 32-kHz sleep timer capture
- CSMA / CA hardware support
- Accurate digital RSSI / LQI support
- 6 channels can be configured 12-bit ADC resolution
- AES security coprocessor
- 2 Powerful USART interface to support multiple serial protocol
- 1 common SPI interface
- 8 general purpose I / O pins (6 × 4 mA, 2 × 20 mA)
- Watchdog Timer
- Wireless Performance
  - Adapt 2.4GHz IEEE802.15.4 RF transceiver
  - High receiver sensitivity and robustness
  - Programmable output power up to 4.5dBm
- Low power consumption
  - Active mode RX (CPU idle): 29mA
  - TX Active mode in 1dBm (CPU idle): 29mA
  - Power mode 1 (4us wake): 0.2mA
  - Power Mode 2 (Sleep timer runs): 0.2mA
  - Power Mode 3 (external interrupts): 0.4uA
  - Wide supply voltage range (2V-3.6V)

2.3 Some of the features TI CC2540/1

- Excellent performance and low power 8051 microcontroller core with code prefetch feature
- 256K system programmable flash memory
- 8KB RAM, with a power supply in a variety of data retention
- Hardware debugging support
- The powerful 5-channel DMA
- Extended baseband automation, including automatic recognition and address decoding
- General-purpose timers (one 16bit, two 8bit)
- IR generating circuit

- 32-kHz sleep timer capture
- Accurate digital RSSI / LQI support
- 2 channels can be configured 12-bit ADC resolution
- AES security coprocessor
- 2 Powerful USART interface to support multiple serial protocol
- 1 common SPI interface
- 8 general purpose I / O pins (6 × 4 mA, 2 × 20 mA)
- Watchdog Timer
- Wireless Performance

  Suitable excellent receiver sensitivity (at 1 Mbps is -94 dBm), selectable, and barrier properties

  Programmable output power up to 4dBm (2540) / 0dBm (2541)
- 2541 Low Power

  RX mode low: 17.9 mA

  Mode TX (0 dBm): 18.2 mA

  Power mode 1 (4- μ s wake): 270 μ A

  Power Mode 2 (Sleep timer open): 1 μ A

  Power Mode 3 (external interrupts): 0.5 μ A

  Wide supply voltage range (2 V-3.6 V)
- 2540 Low Power

  RX mode low: 19.6 mA

  Mode TX (-6 dBm): 24 mA

  Power mode 1 (3- μ s wake): 235 μ A

  Power Mode 2 (Sleep timer Open): 0.9 μ A

  Power Mode 3 (external interrupts): 0.4 μ A

  Wide supply voltage range (2 V-3.6 V)

2.4 Some of the features STM32F103CB

- ARM Cortex-M3 core running at 72MHz
- 128K system programmable flash memory
- 20KB RAM
- 2 × 12 bits 1 microsecond A / D converter (up to 16 channels)

  Conversion range: 0~3.6 V

  Dual sample and hold function
- DMA

  7-channel DMA controller

  Supported peripherals: timers, ADC, SPI's, I2C and USART
- 18 general-purpose programmable GPIO
- 7 Timer

  3 16-bit timers, each up to 4 IC / OC / PWM or pulse counter quadrature (incremental) encoder into

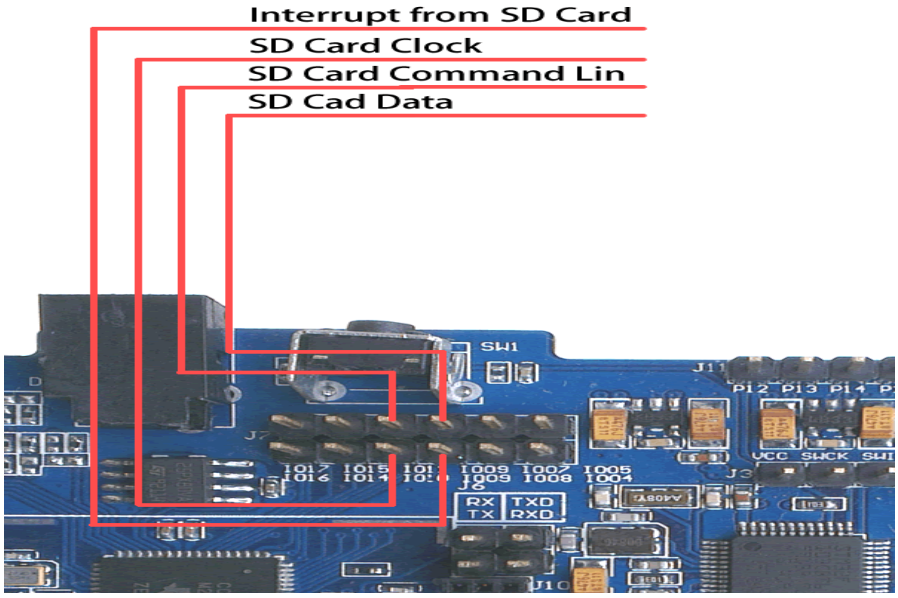
16-bit motor control PWM timer, dead-time generation and emergency stop
2 watchdog timers (Independent and windows)
System Timer: 24 downcounter

- 1 I2C interface (SMBus / PMBus)
- 3 USART (ISO7816 interface, LIN, infrared capabilities, modem control)
- 1 SPI interface (18 Mbit / s)
- 1 CAN interface (2.0B Active)
- 1 full-speed USB 2.0 interface

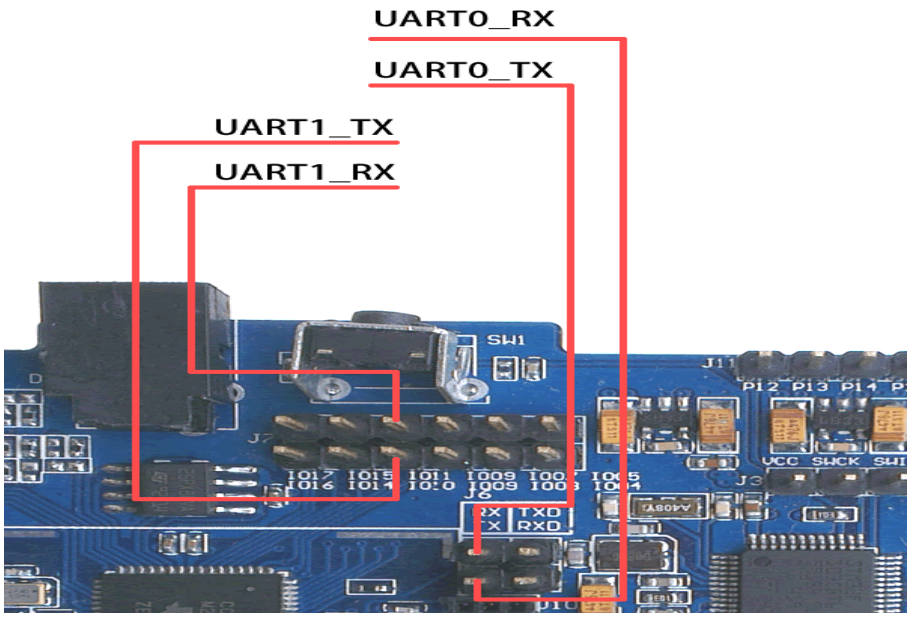
3 Peripheral Interface

3.1 TI CC3200 part Peripheral Interface

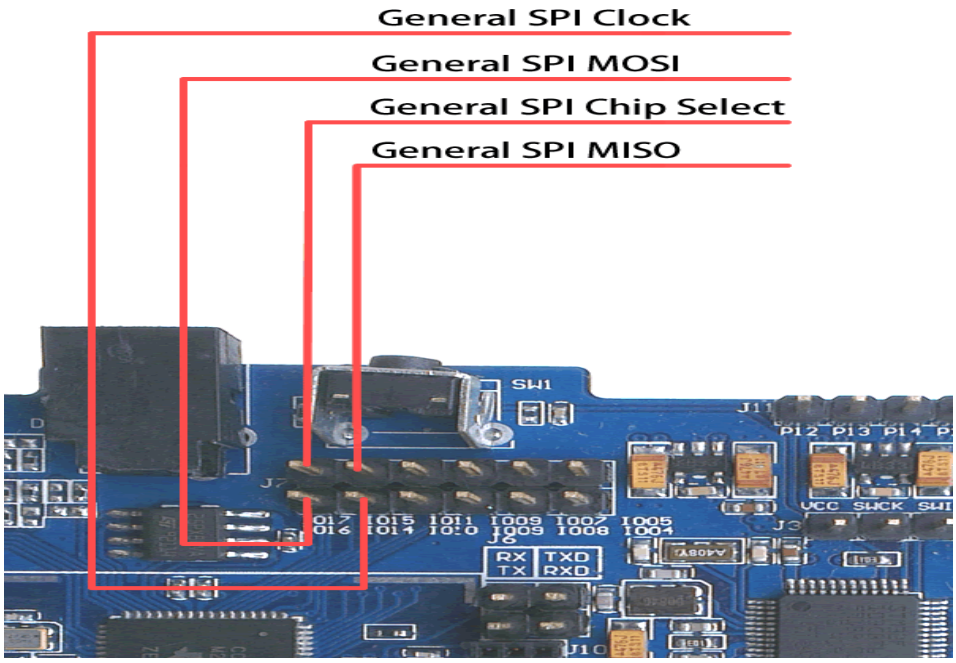
3.1.1 4-wire SDIO Interface



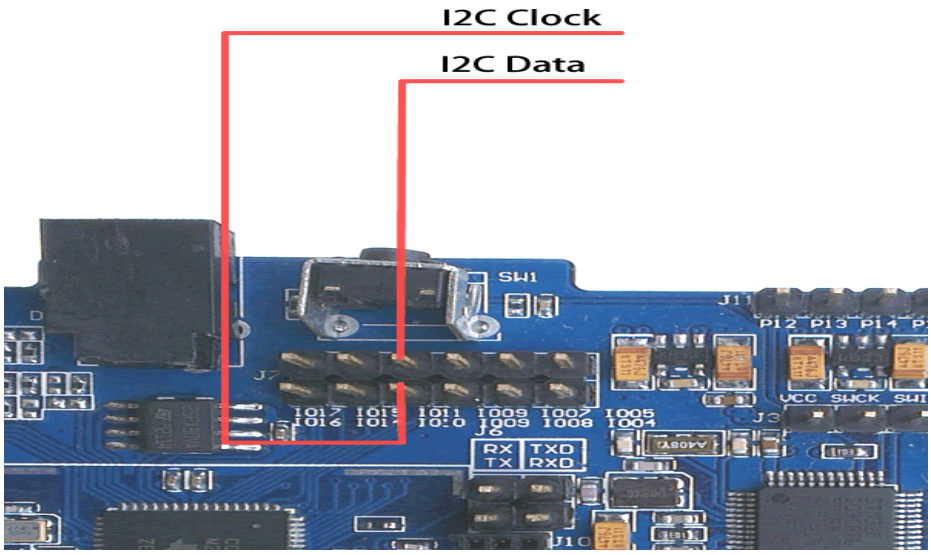
3.1.2 Uart Interface



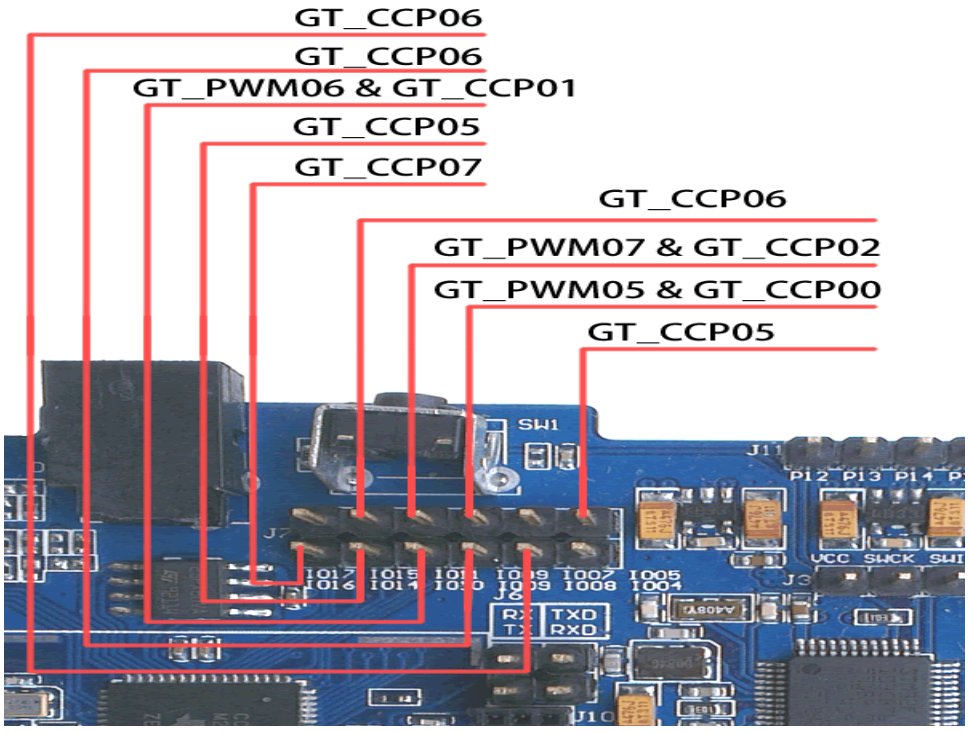
3.1.3 SPI Interface



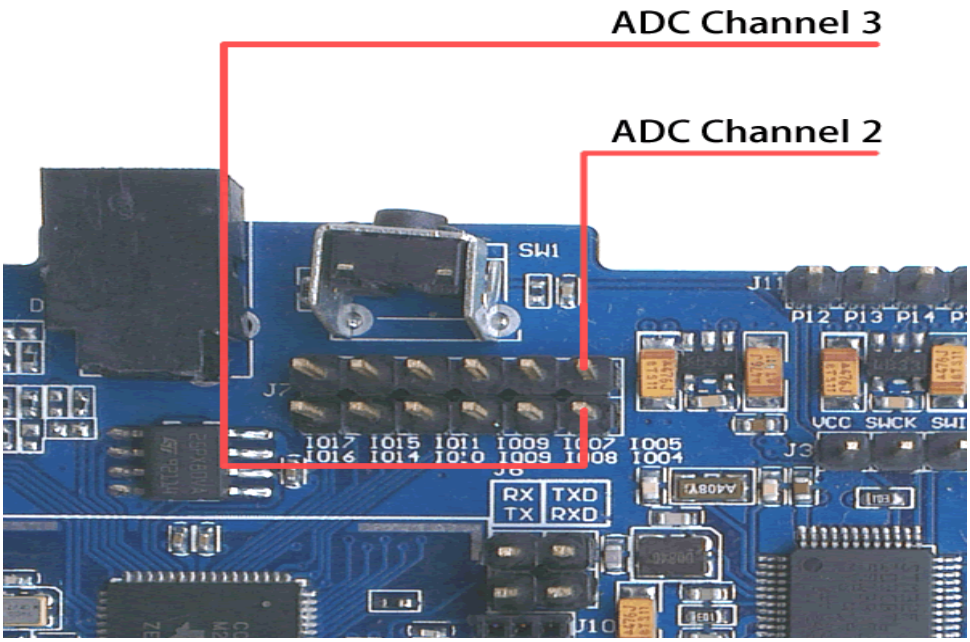
3.1.4 I2C Interface



3.1.5 PWM & Timer Capture channel

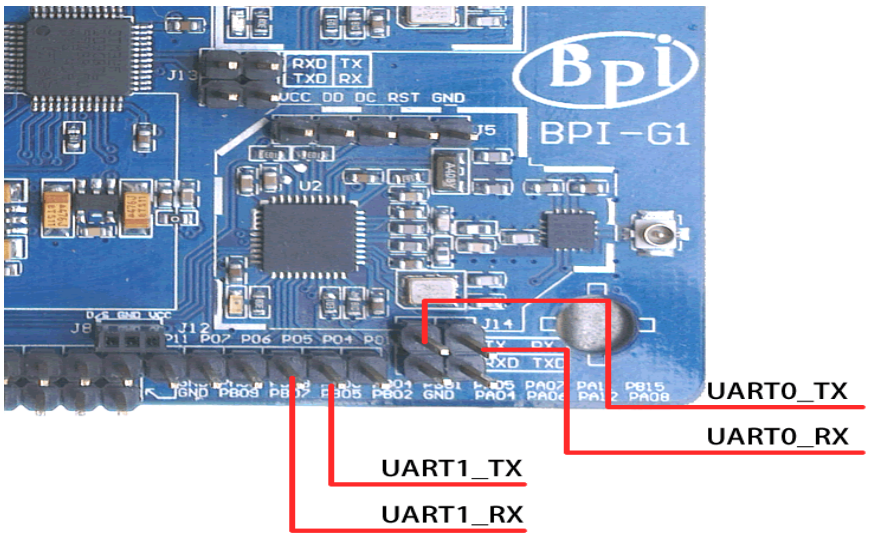


3.16 ADC channel

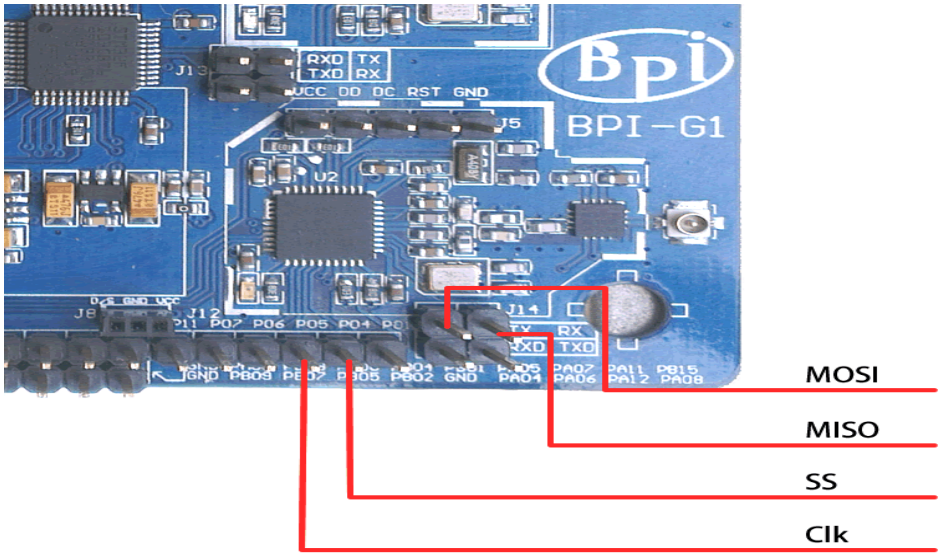


3.2 TI CC2530 part Peripheral Interface

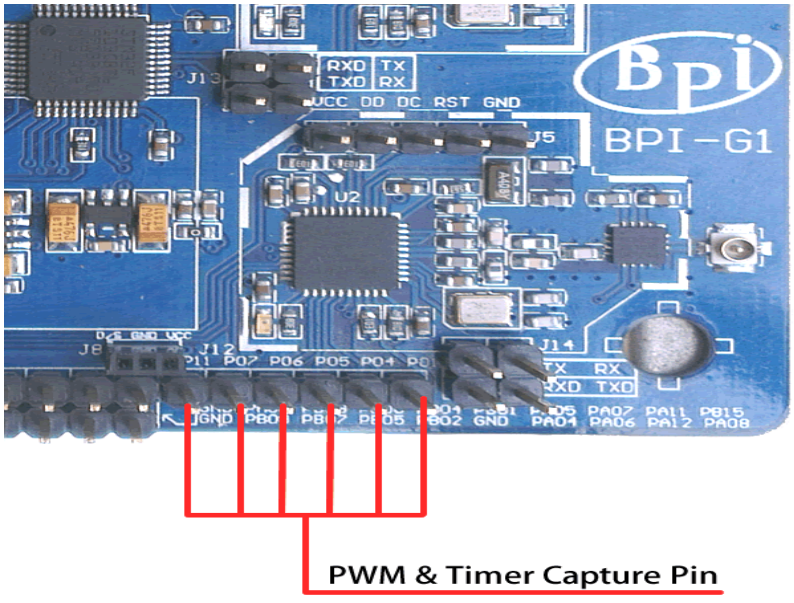
3.2.1 Uart Interface



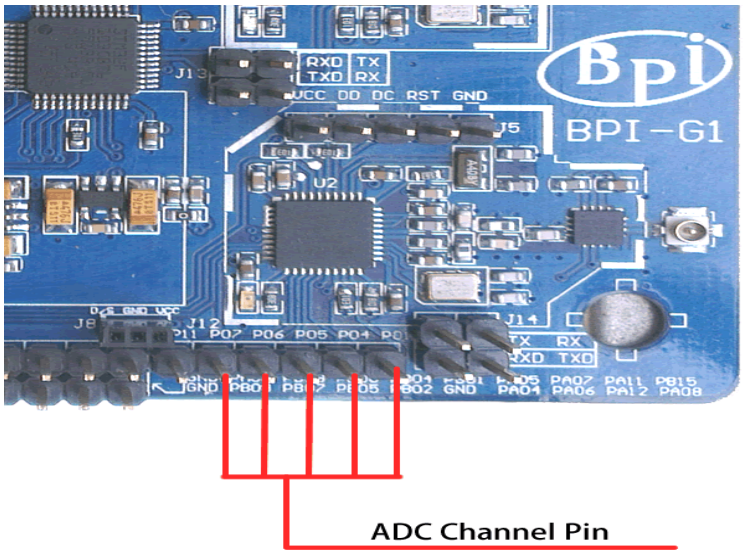
3.2.2 SPI Interface



3.2.3 PWM & Timer Capture channel

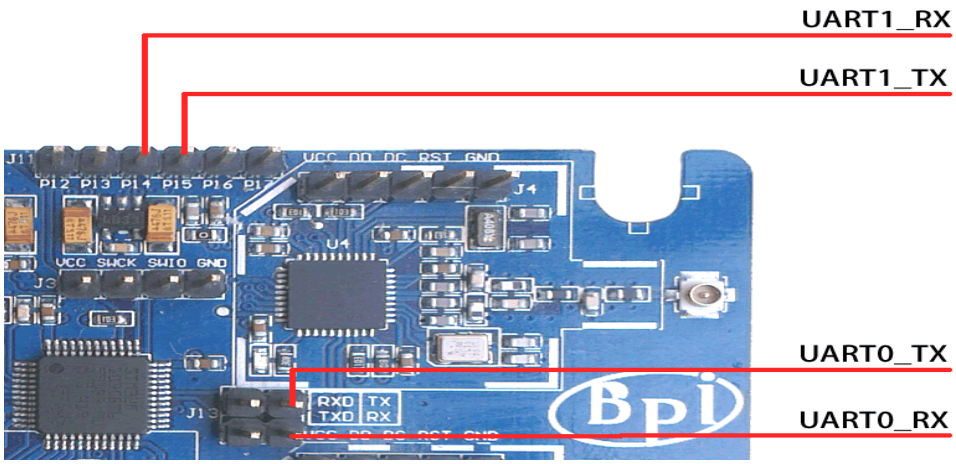


3.2.4 ADC channel

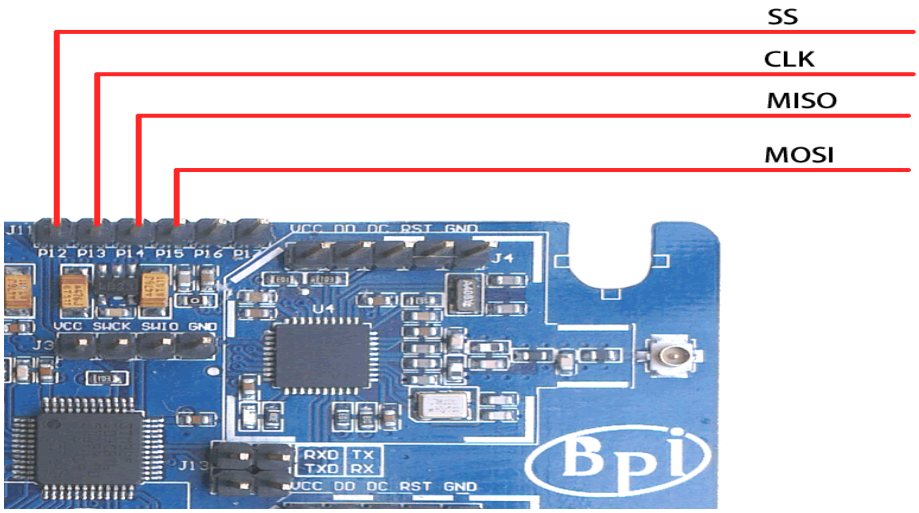


3.3 TI CC2540/1 part Peripheral Interface

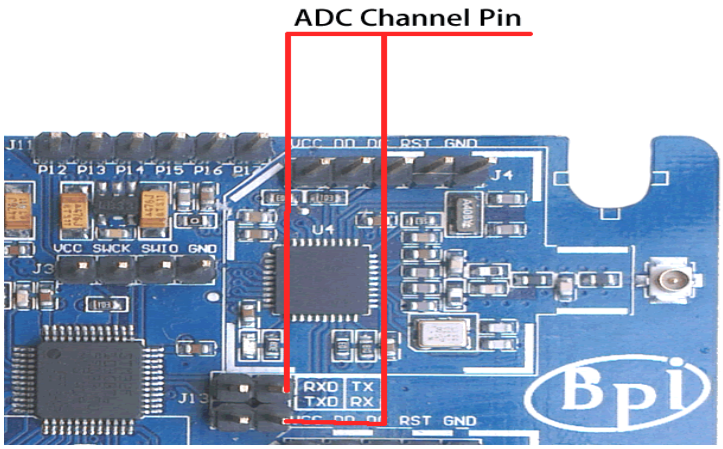
3.3.1 Uart Interface



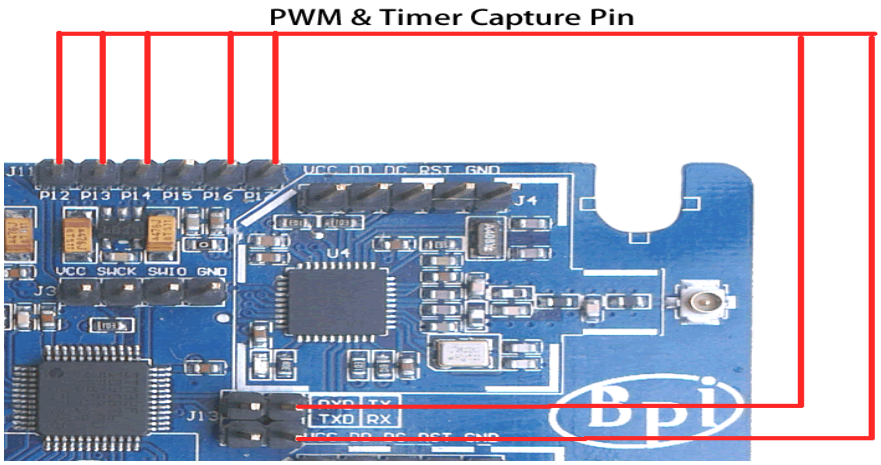
3.3.2 SPI Interface



3.3.3 ADC channel

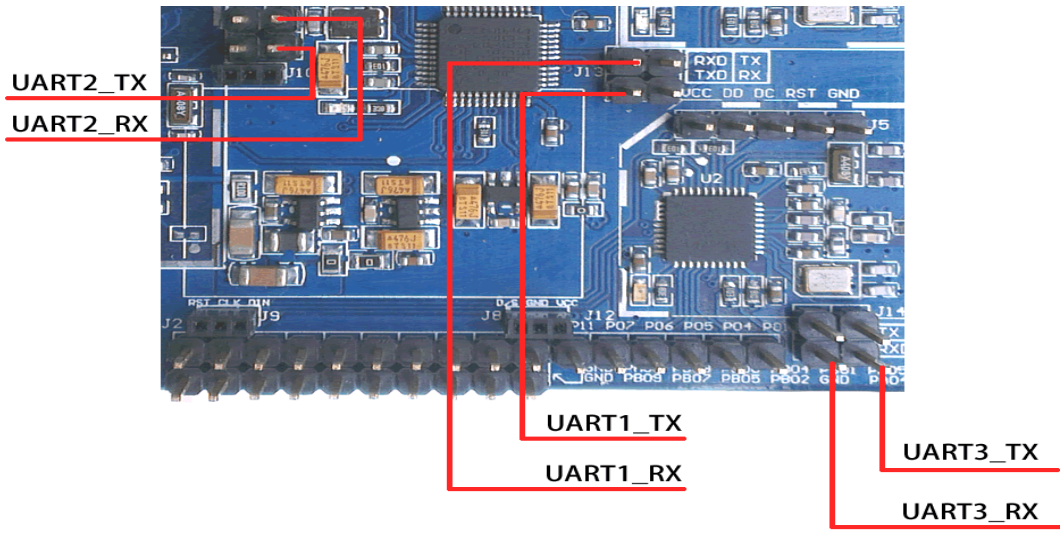


3.3.4 PWM & Timer Capture channel

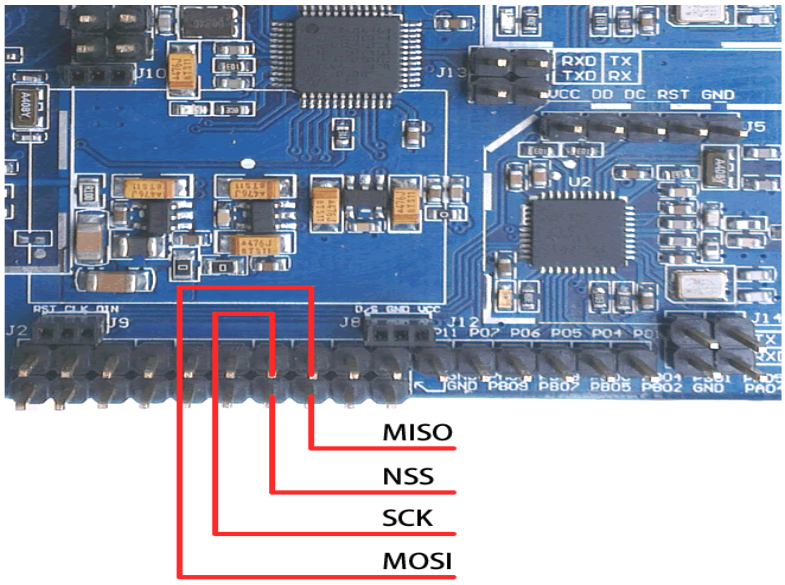


3.4 STM32F103CB part Peripheral Interface

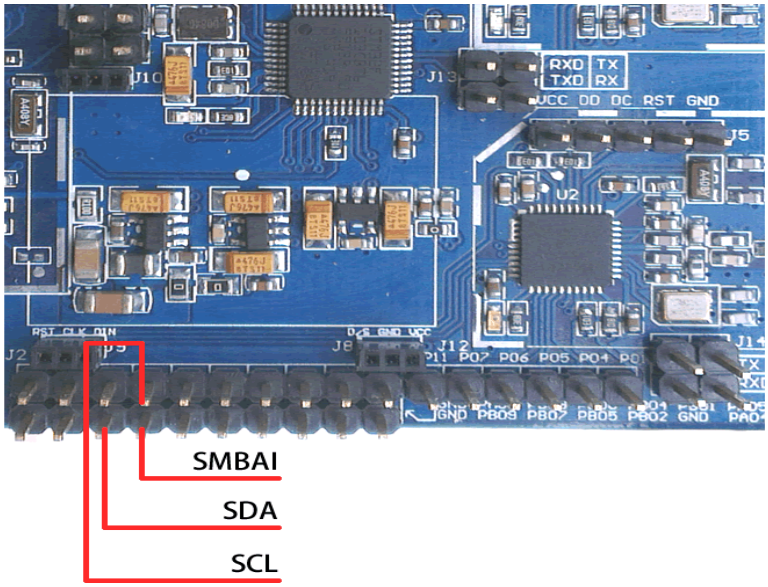
3.4.1 Uart Interface



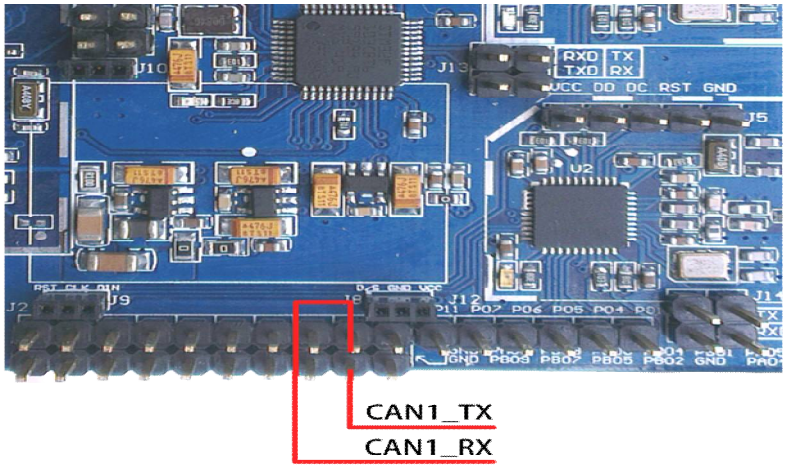
3.4.2 SPI Interface



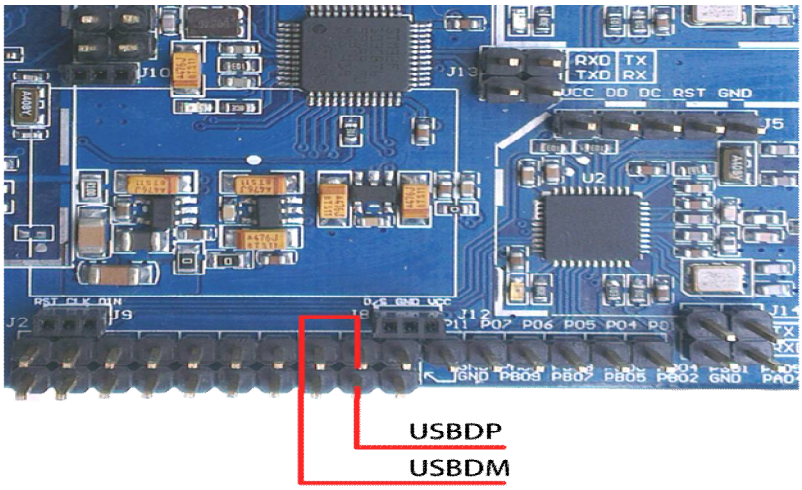
3.4.3 I2C Interface



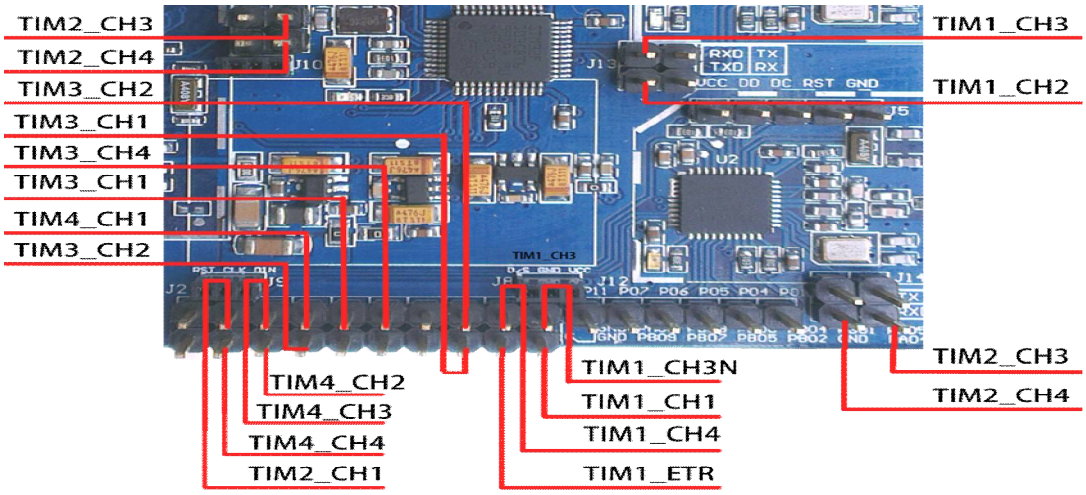
3.4.4 CAN Interface



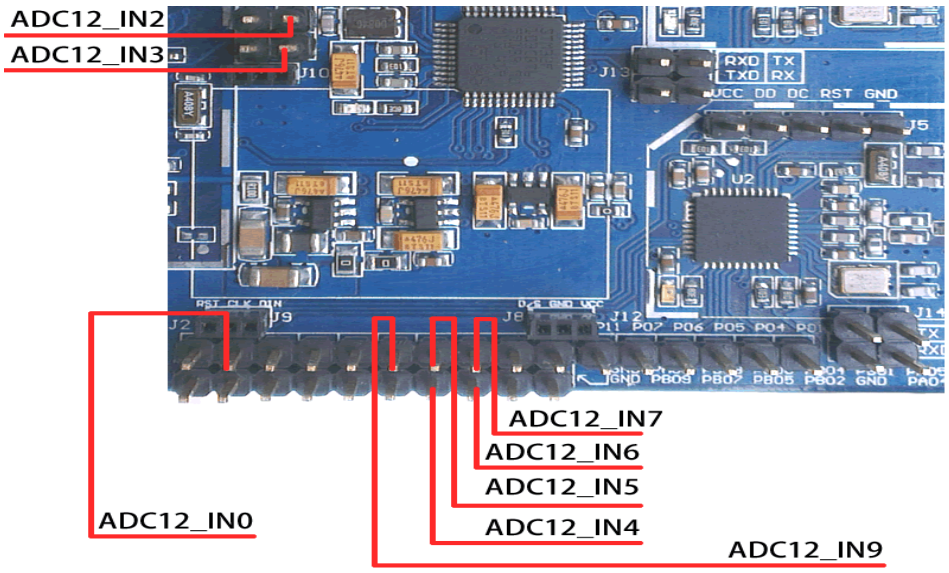
3.4.5 USB2.0 Interface



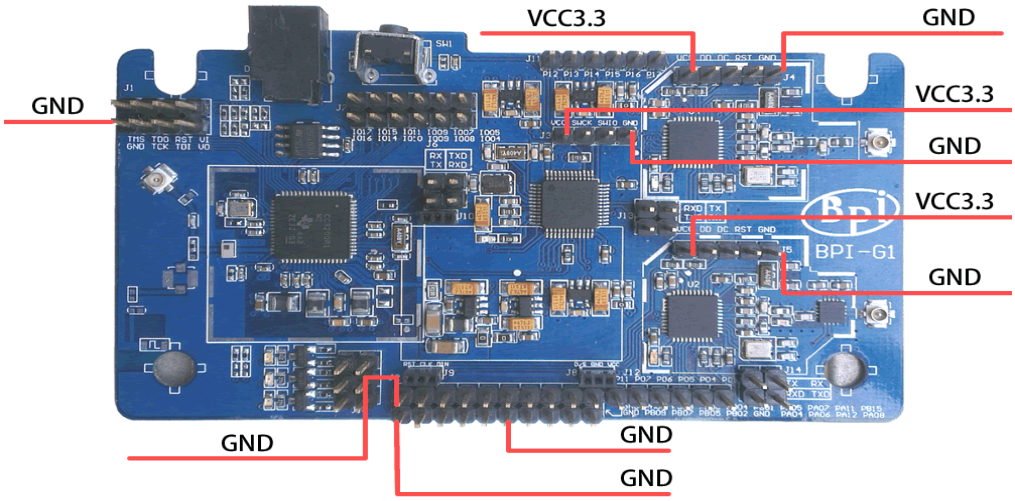
3.4.6 PWM & Timer Capture channel



3.4.7 ADC channel

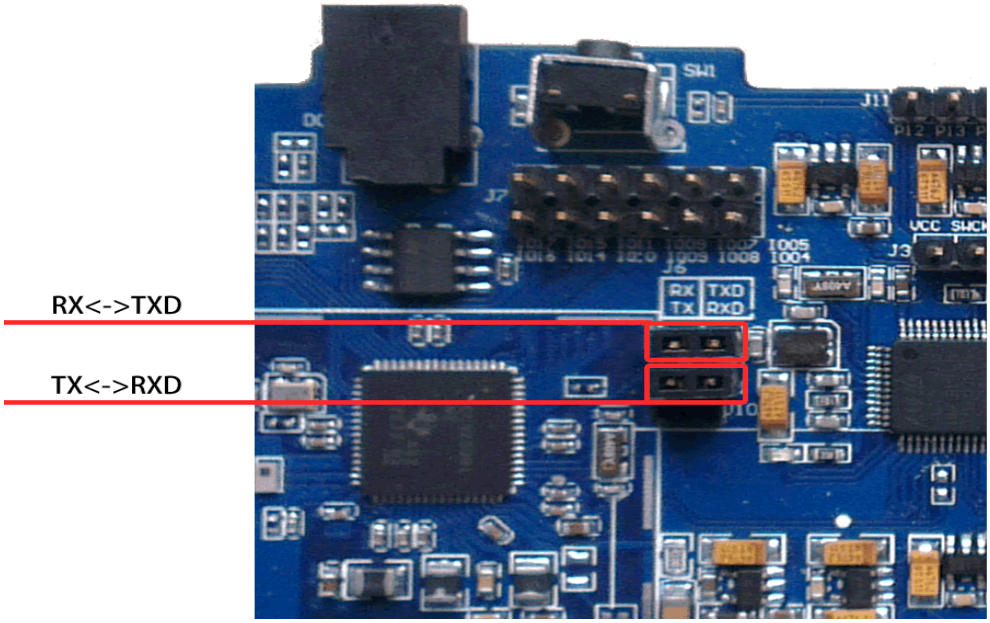


4 Global Power and Ground

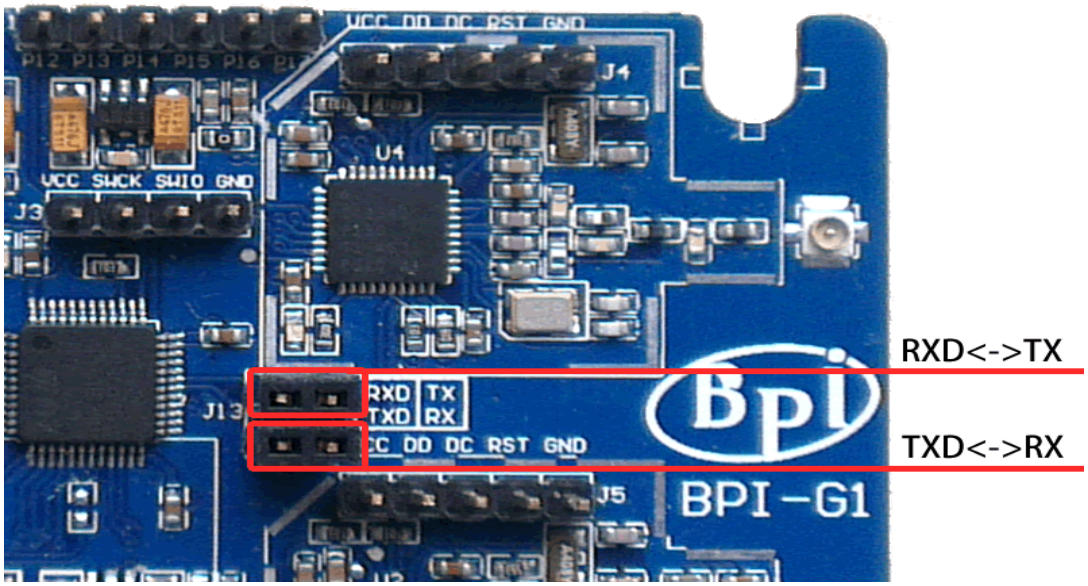


5 Module interconnection

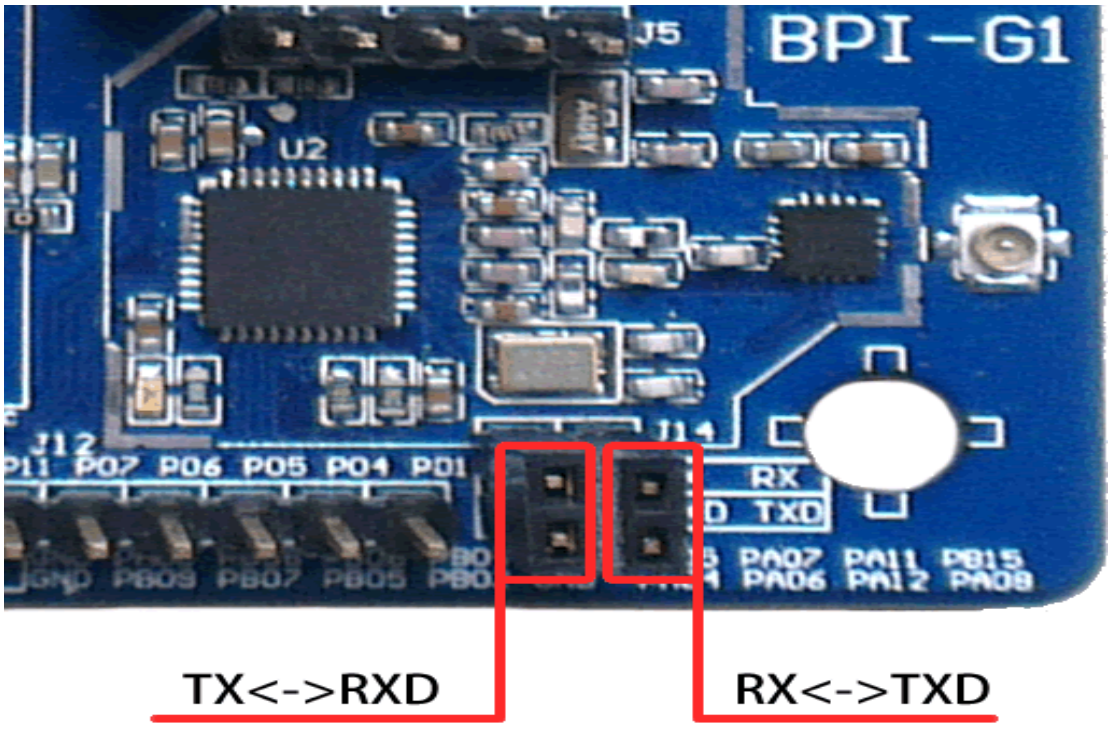
WIFI (CC3200), BLE (CC2540 / 1) and ZIGBEE (CC2530) are interconnected through STM32F103CB serial port, they can work alone, can also coordinate the work will be given below communication connections between the modules schematic。5.1 CC3200 connection with F103



5.2 CC2540 connection with F103

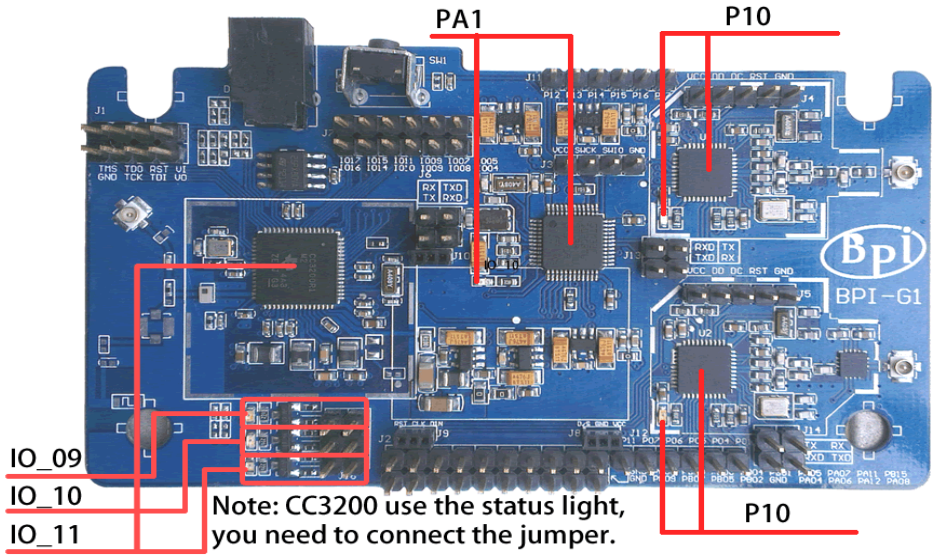


5.3 CC2530 connection with F103



6 Status LEDs for each module

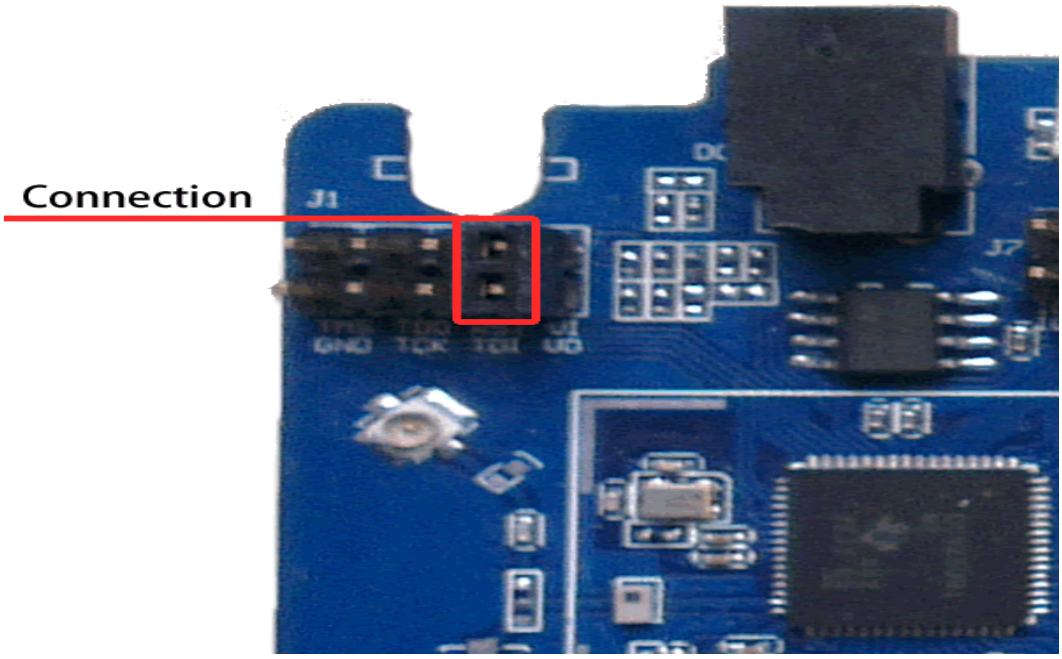
In each module has its own status indicators, in addition there are three status lights CC3200, the others were all one, and these status lights are controlled by the GPIO module。



7 TI CC3200 mode selection jumper

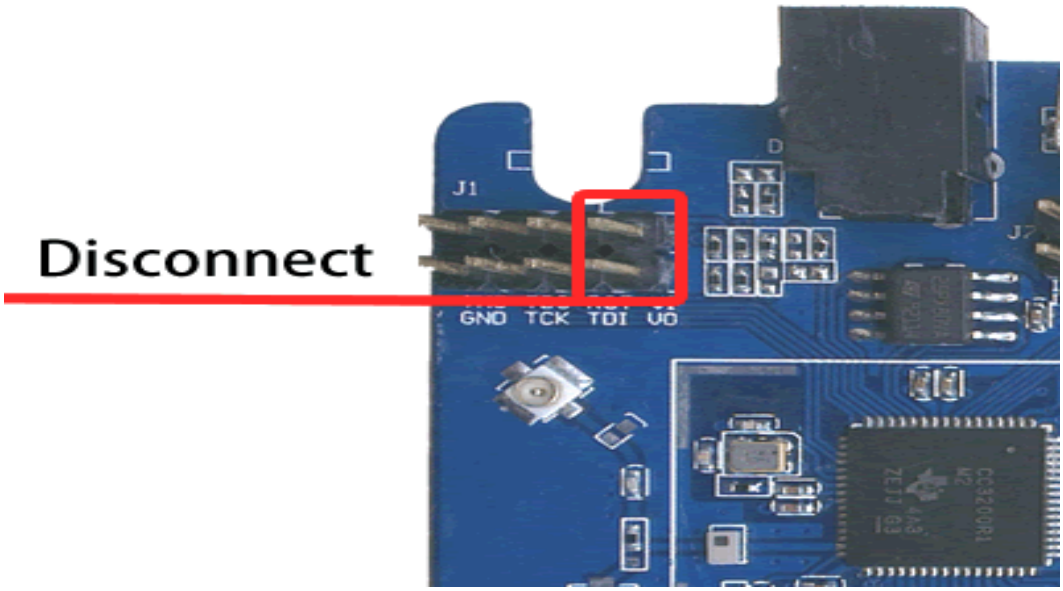
7.1 Programming mode

Under the programming mode, you need to VI and VO short on power, CC3200 will receive as a writer, it will be the serial data received by the SPI mode file system writes to the external SFLASH, but the programming is completed after not running, you need to disconnect VI and VO, then on again, then it will be read in the file system SFLASH into memory and execute code. As shown in Figure。



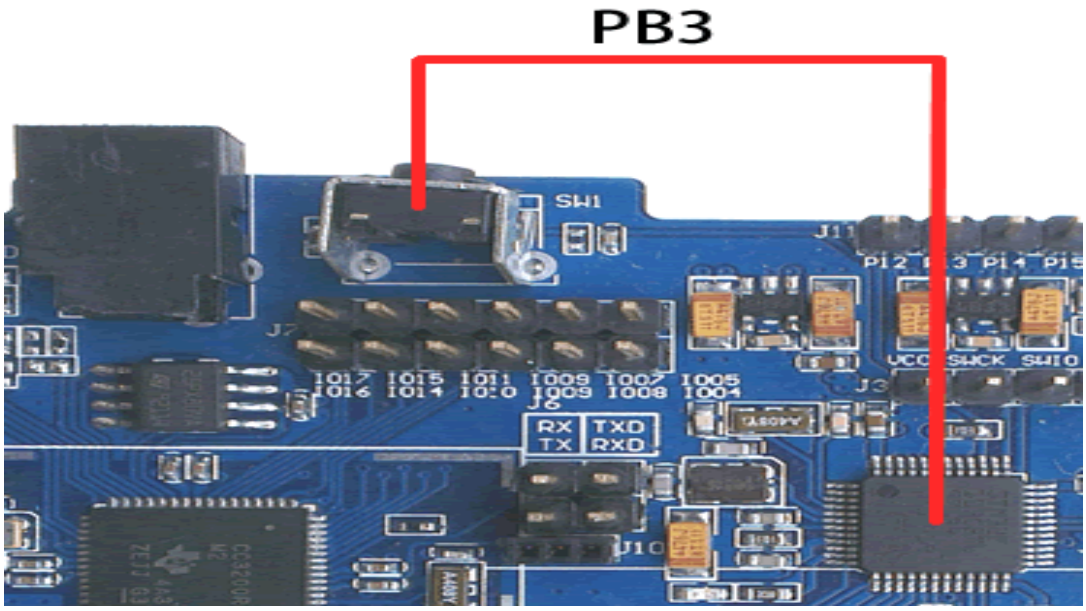
7.2 Run mode

Run mode, you need to disconnect VI and VO, then power on, CC3200 will SFLASH read from the file system into memory and run. In this mode, plug in the JTAG interface, using the IDE's DEBUG function, and the code will be downloaded from the JTAG interface to the memory to run, you can perform single-step debugging. For fast functional verification, this is the fastest way. However, after a reset circuit, CC3200 will be new to SFLASH read the file system operation. As shown in Figure。



8 Button

BPI G1 has a button, which is connected to the PB3 pin F103, in the middle of a pullup resistor in parallel, when the key is pressed, the pin will be low-level information. Figure。



9 How to Quick Start?

Demonstration IDE, here I will use the IAR EWARM under Windows7 and IAR EW8051, debugging emulator will use BPI OPENDEBUG.。

9.1 Quick start of TI CC3200

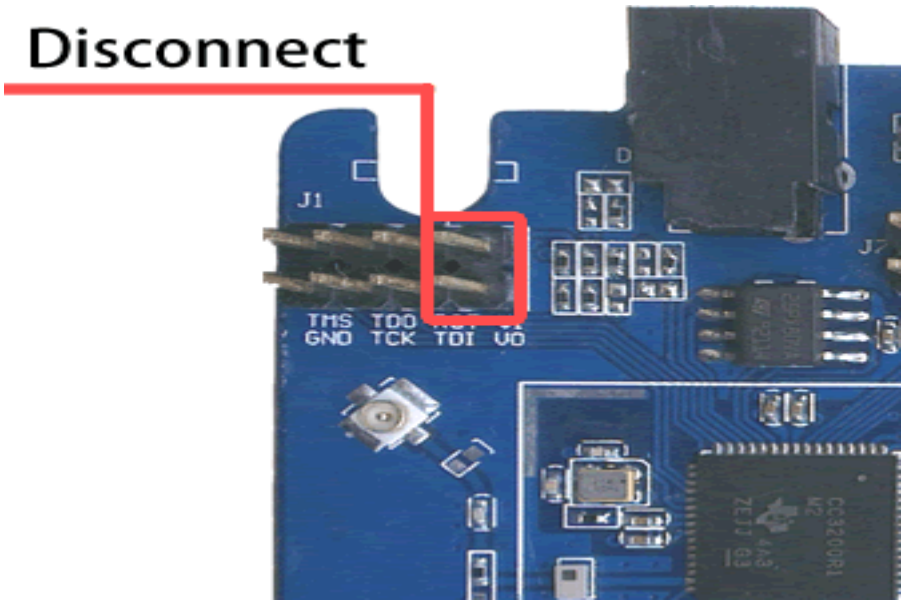
9.1.1 Download and install the software

Download and install the following software：

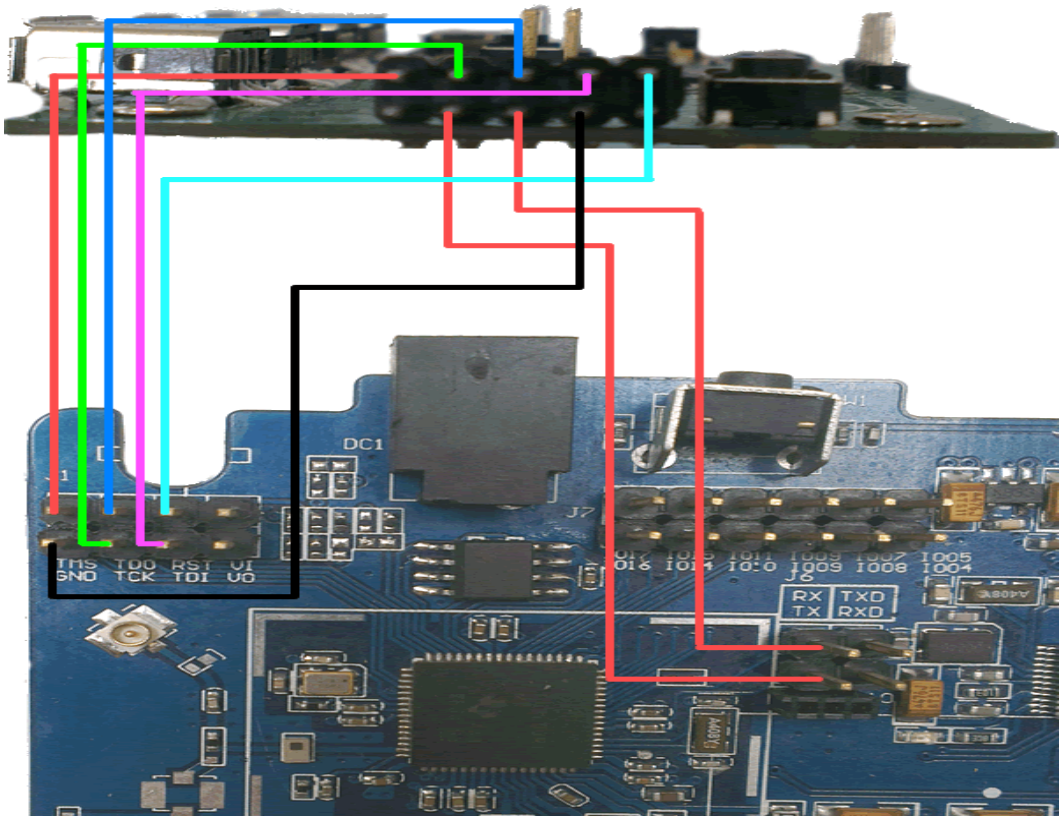
- CC3200 SDK Package
  - Assuming the default installation folder C:\ TI\ CC3200SDK_1.0.0\
- IAR EWARM（Please install the software on their own, we recommend using IAR EWARM 7.2 or later）

- Tera Term (or similar software)
  - Tera Term link: http://en.sourceforge.jp/projects/ttssh2/releases/

9.1.2 Configuration G1

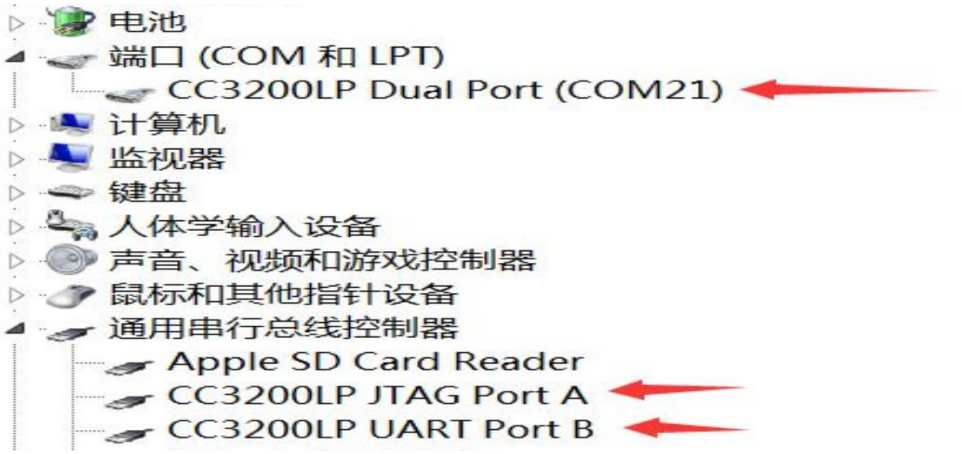


9.1.3 G1 connection BPI OPENDEBUGGER



9.1.4 Check whether the connected BPI OPENDEBBUG

BPI OPENDEBUGGER connected to the PC, and check whether the drive is installed, if you can not identify, install TI UNIFLASH, the programming software that comes with the driver. If you can identify that, please give G1 on electricity。

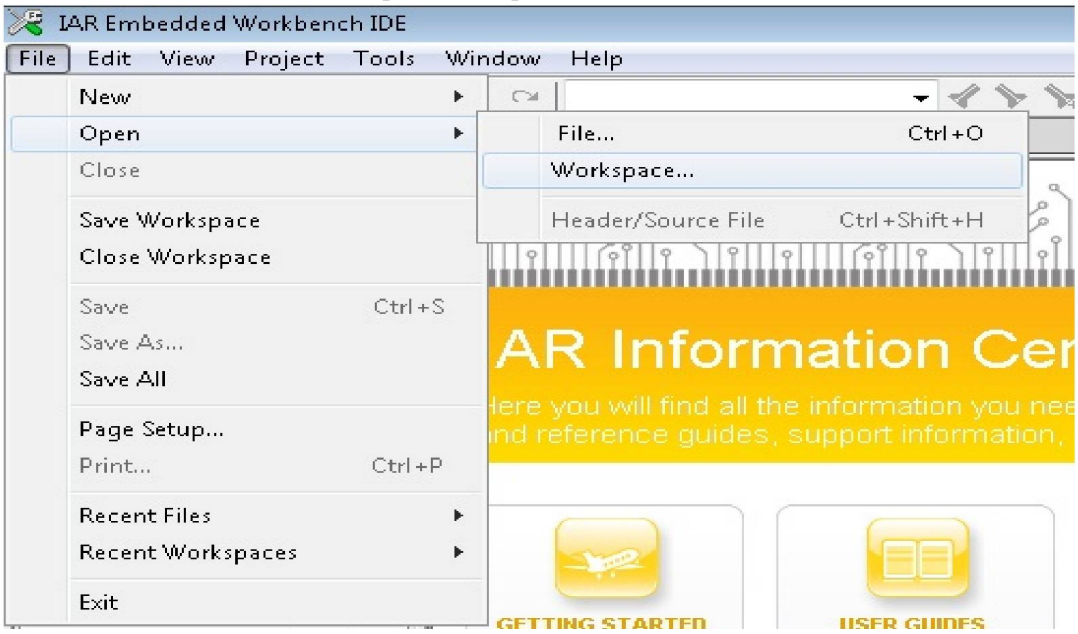


9.1.5 Cover IAR debugger interface file

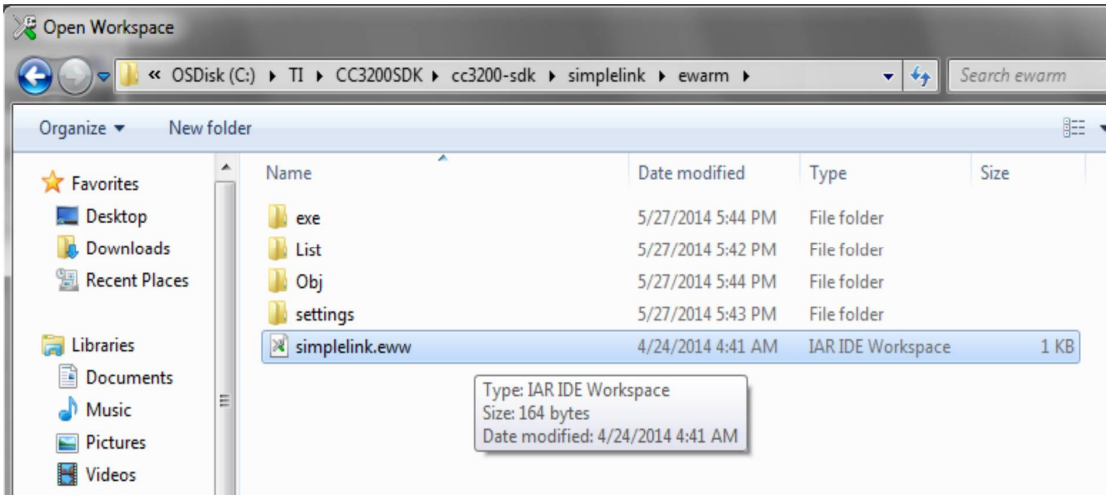
Copy the file C:\TI\CC3200SDK_1.0.0\CC3200-sdk\tools\iar_patch\armLMIFTDI.dll to folder C:\Program Files (x86)\IAR Systems\Embedded Workbench 7.0\arm\bin (Select be covered).

9.1.6 Recompile SimpleLink library files

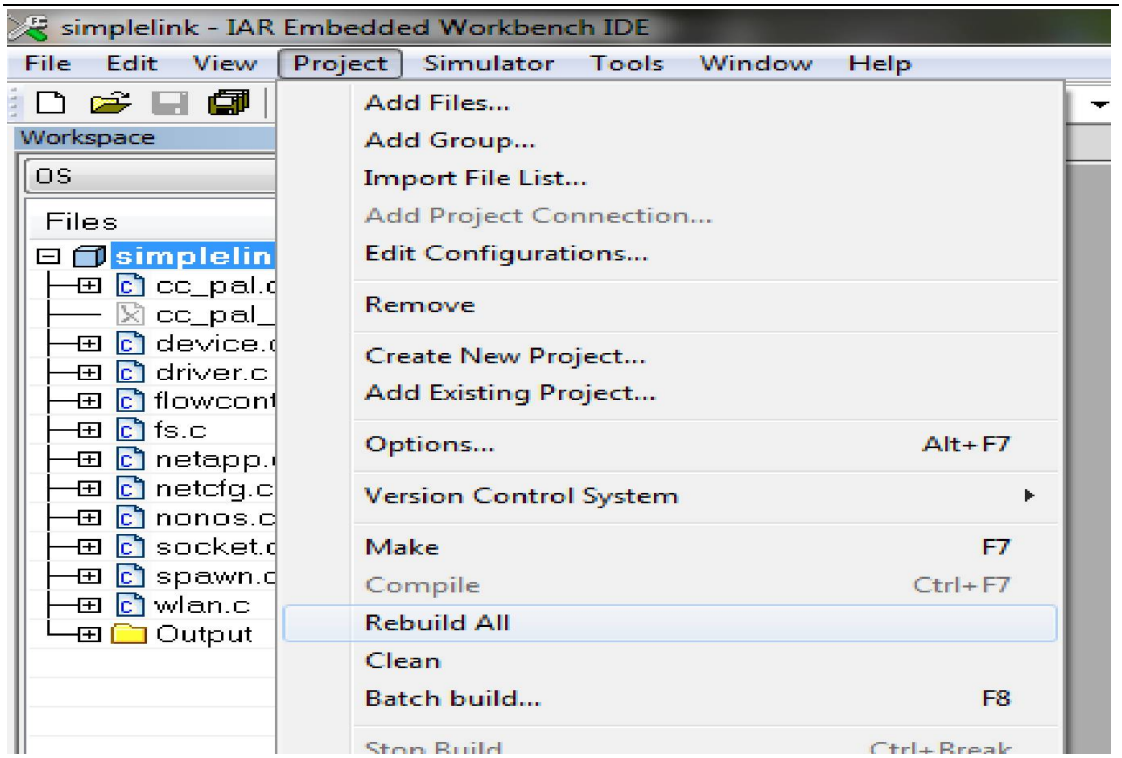
1. Start the IAR and select File>Open>Workspace from the menu..



2. Browse C:\TI\CC3200SDK_1.0.0\cc3200-sdk\simplelink\ewarm directory and open simplelink.eww



3. From the menu, select Project>Rebuild All to Recompile simplelink project

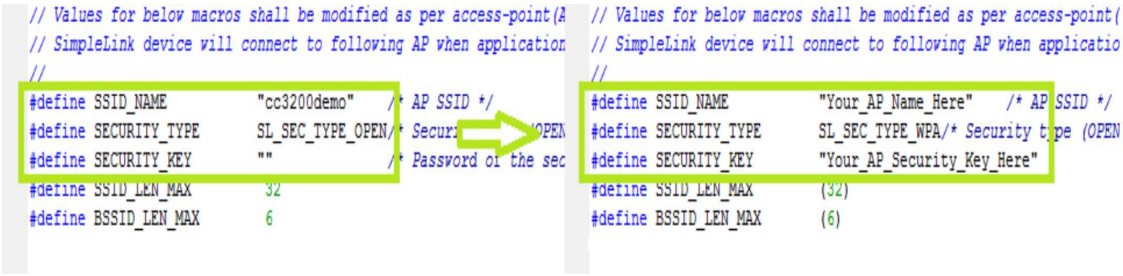


9.1.7 Compile, download and debug routines WLAN Station

1. Open the wlan_station project by selecting File>Open>Workspace from the menu, navigating to C:\TI\CC3200SDK_1.0.0\cc3200-sdk\example\getting_started_with_wlan_station\ewarm, and opening wlan_station.eww。

2. Open the common.h file located at the path C:\TI\CC3200SDK_1.0.0\cc3200-sdk\example\common\。

3. Edit common.h to use the SSID, security type and security key of the Access Point being used. Edit the macros SSID_NAME, SECURITY_TYPE and SECURITY_KEY to contain the Access Point's information as shown in Figure. The security types supported for this demo are WPA/WPA2 and Open. For Open security, define SECURITY_TYPE as SL_SEC_TYPE_OPEN. For WPA and WPA2 security, define it as SL_SEC_TYPE_WPA。

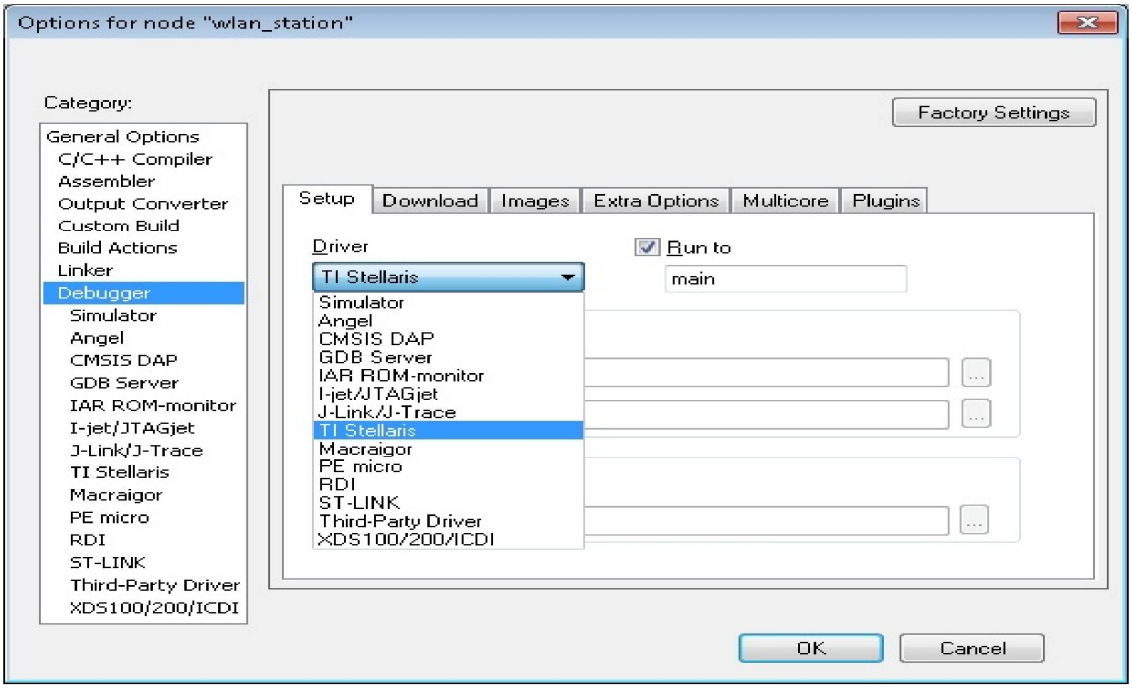

// Values for below macros shall be modified as per access-point(A
// SimpleLink device will connect to following AP when application
//
#define SSID_NAME "cc3200demo" /* AP SSID */
#define SECURITY_TYPE SL_SEC_TYPE_OPEN/* Securi
#define SECURITY_KEY "" /* Password of the sec
#define SSID_LEN_MAX 32
#define BSSID_LEN_MAX 6
// Values for below macros shall be modified as per access-point(
// SimpleLink device will connect to following AP when applicatio
//
#define SSID_NAME "Your_AP_Name_Here" /* AP SSID */
#define SECURITY_TYPE SL_SEC_TYPE_WPA/* Security type (OPEN
#define SECURITY_KEY "Your_AP_Security_Key_Here"
#define SSID_LEN_MAX (32)
#define BSSID_LEN_MAX (6)

4.Save common.h。

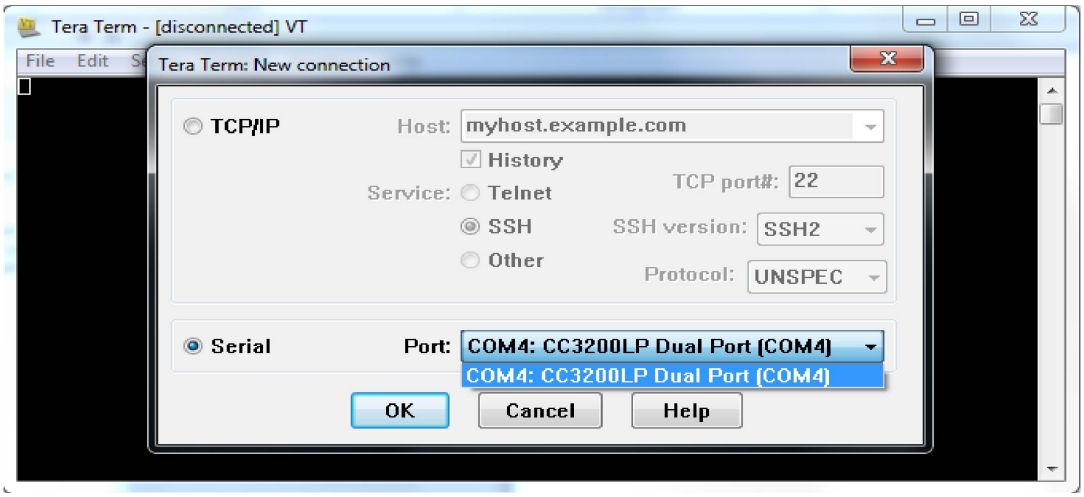
5.Rebuild the wlan_station project by selecting Project>Rebuild All from the menu。

6.The debugger must be configured to download code to the device. Select Project>Options from the menu, and select the Debugger category. In the Setup tab, choose TI Stellaris as the driver, as

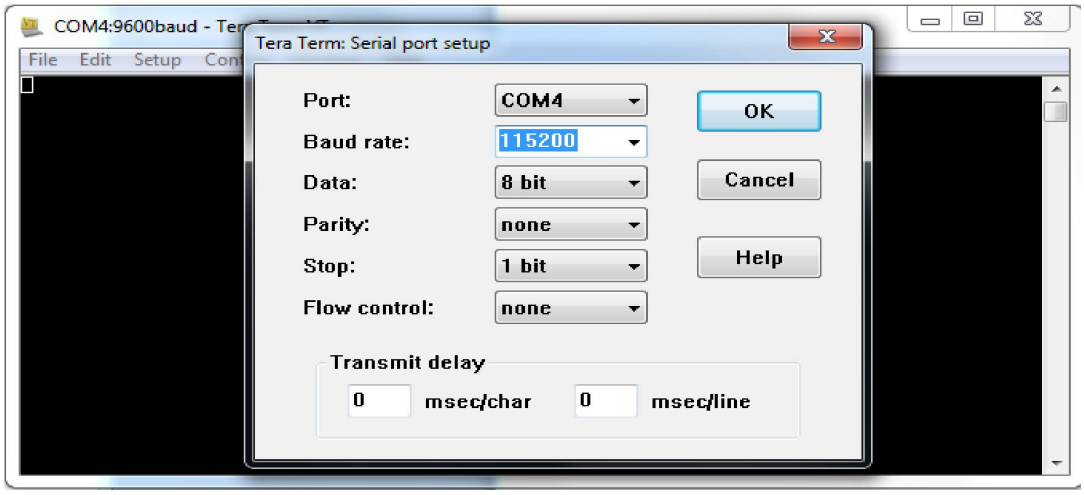
shown in Figure, and press OK。



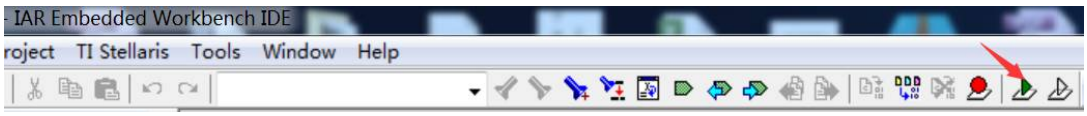
7.Launch Tera Term, and create a new serial connection to the CC3200 Launchpad COM port as shown in Figure。



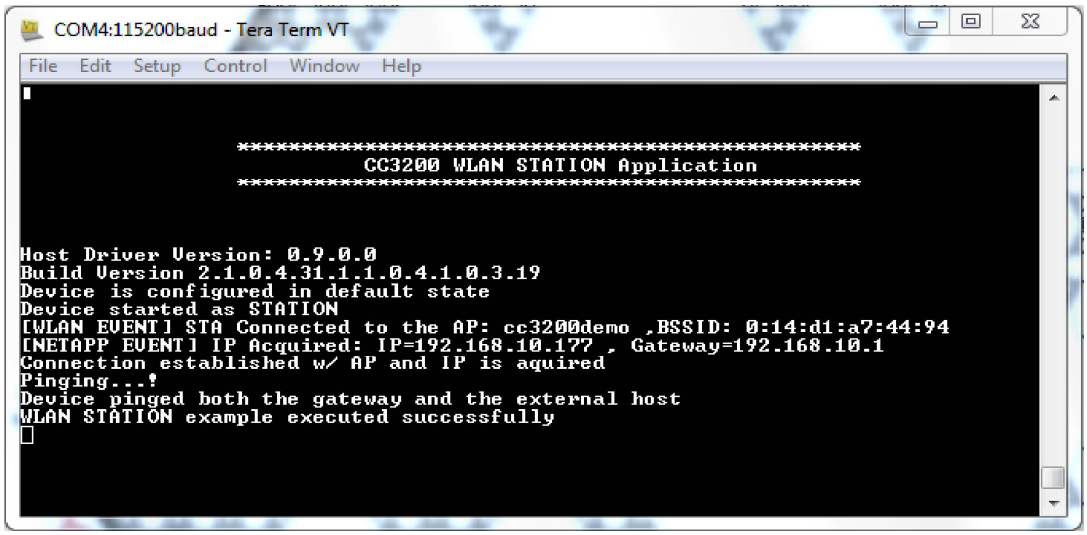
8.In the menu, select Setup>Serial Port, and change the baud rate to 115200 as shown in Figure。



9. Click the debug icon as shown in Figure 24 to download code to the device and start debugging.Select Debug>Go from the menu or press F5 to begin execution



10.If the CC3200 successfully completes all steps, the serial output appears as shown in Figure。


COM4:115200baud - Tera Term VT
File Edit Setup Control Window Help
CC3200 WLAN STATION Application
Host Driver Version: 0.9.0.0
Build Version 2.1.0.4.31.1.1.0.4.1.0.3.19
Device is configured in default state
Device started as STATION
[WLAN EVENT] STA Connected to the AP: cc3200demo ,BSSID: 0:14:d1:a7:44:94
[NETAPP EVENT] IP Acquired: IP=192.168.10.177 , Gateway=192.168.10.1
Connection established w/ AP and IP is aquired
Pinging...!
Device pinged both the gateway and the external host
WLAN STATION example executed successfully

9.2 Quick start of TI CC2540/1

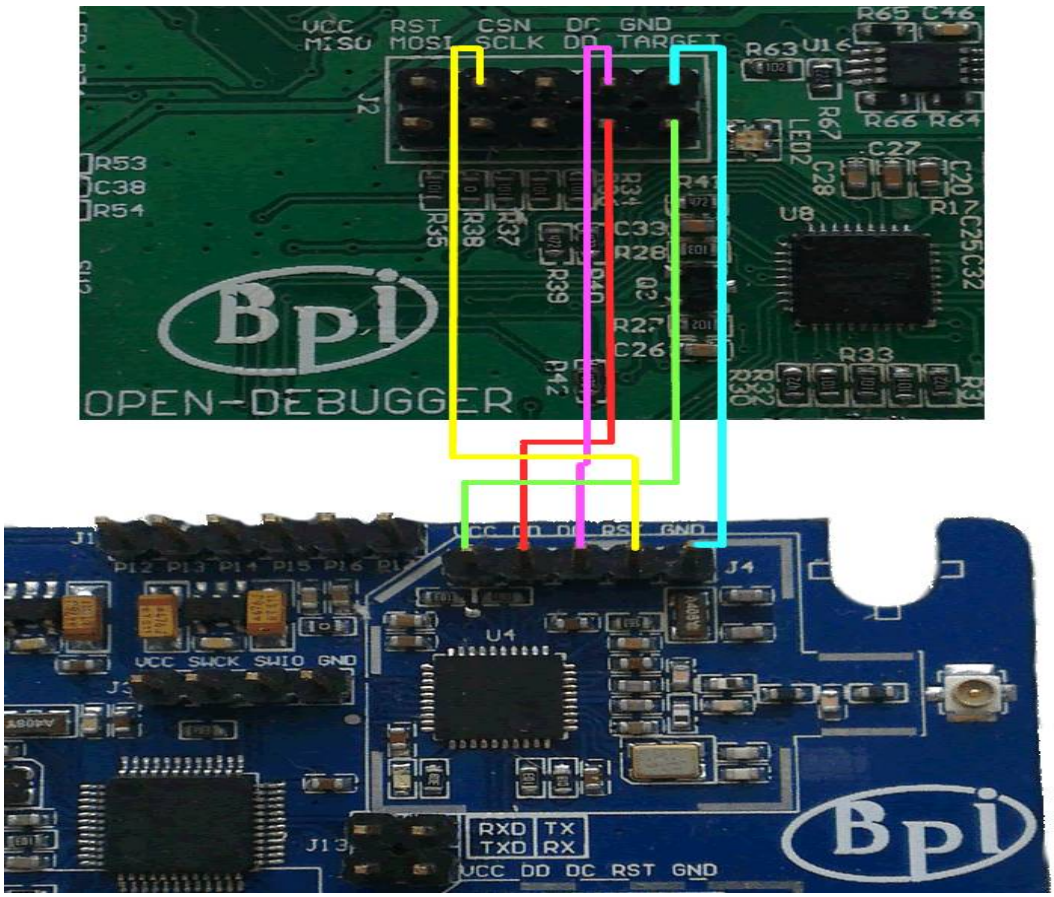
The following operations are by CC2540 prevail, if G1 above your hands BLE chip CC2541, so in some way, your own modifications by CC2541。

9.2.1 Download and install the software

Download and install the following software：

- BLE Stack（Recommends using version 1.3.2, can go to google search to）
  - Assuming the default installation folder C:\Texas Instruments\BLE-CC254x-1.3.2
- IAR EW8051（Please install the software on their own, we recommend using IAR EW8051 8.1）

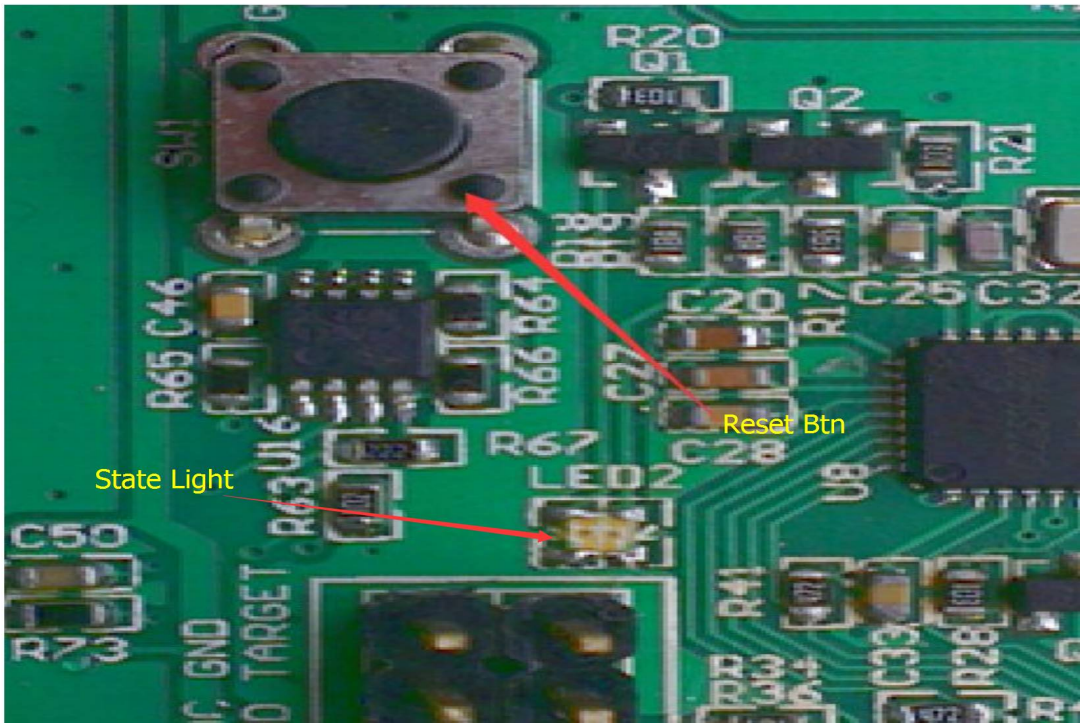
9.2.2 G1 connection BPI OPENDEBUGGER



9.2.3 BPI OPENDEBBUG check whether the software and hardware are connected

BPI OPENDEBUGGER connected to the PC, and check whether the drive is installed, if you can not identify, install SmartRF Studio, he software that comes with the driver. If you can identify that, please give G1 on electricity.

If the drive is normal, then check the G1 and OPENDEBBUG connection is normal. After the G1 to power, press the CC Debugger interface OPENDEBBUG next reset button.。

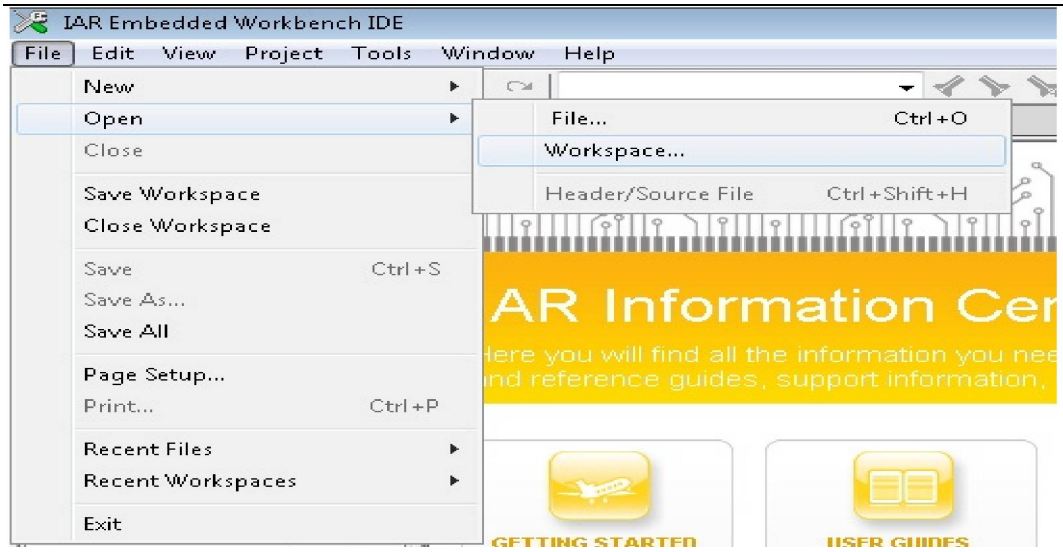


If the connection is normal, then the status light next to the CC Debugger interface will turn green, if it is red or other colors, please check whether it is normal or electricity on the G1 G1 is properly connected and OpenDebugger。

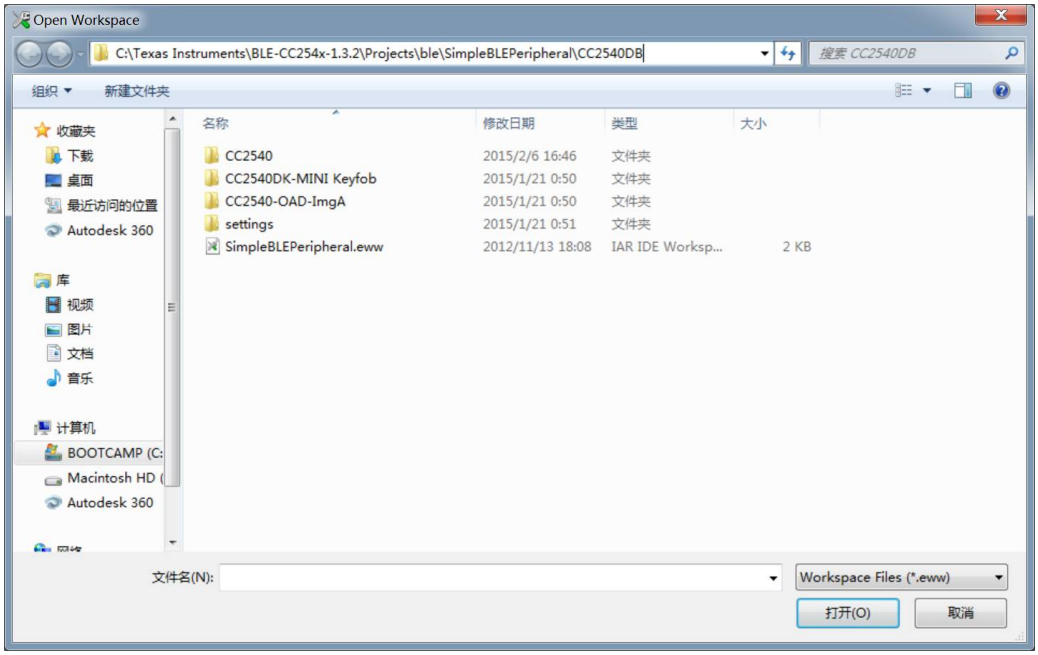
9.2.4 Compile, download and debug routines

SimpleBLEPeripheral

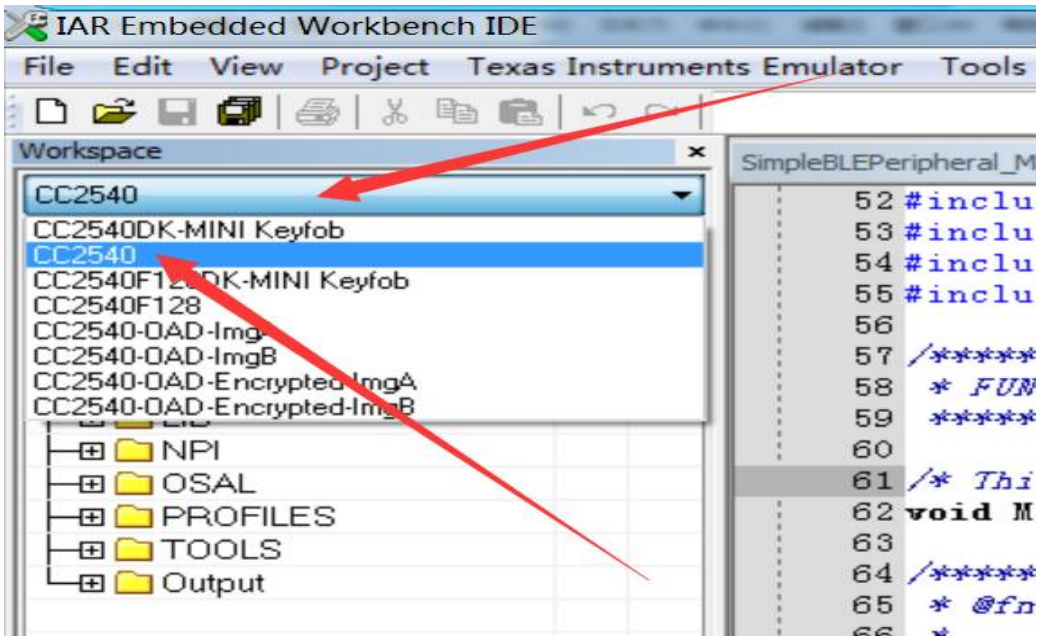
1.IAR EW8051 SimpleBLEPeripheral start and open the project, select File> Open from the menu> Workspace。



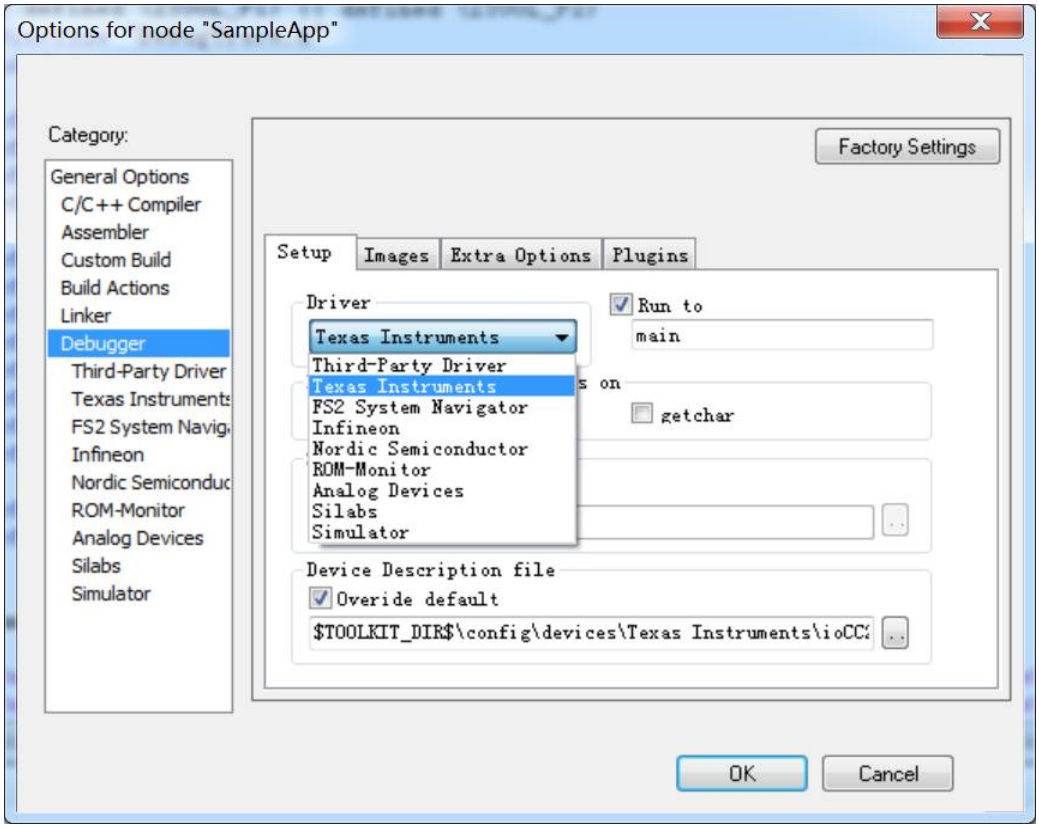
2.Browse to C:\Texas Instruments\BLE-CC254x-1.3.2\Projects\ble\SimpleBLEPeripheral\CC2540DB and open SimpleBLEPeripheral.eww。



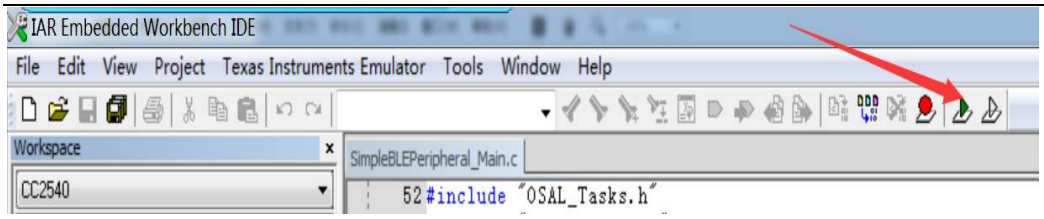
3.Choose from the list of items CC2540。



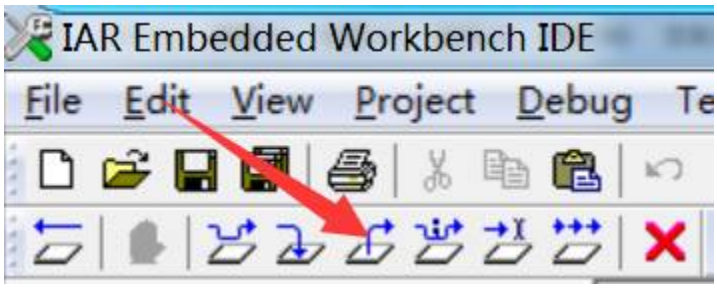
4.When debugging code must be downloaded to the device. Select Project> Options from the menu and select Debugger category. In this setup tab, select Texas Instruments as a driver, and then click OK. Figure。



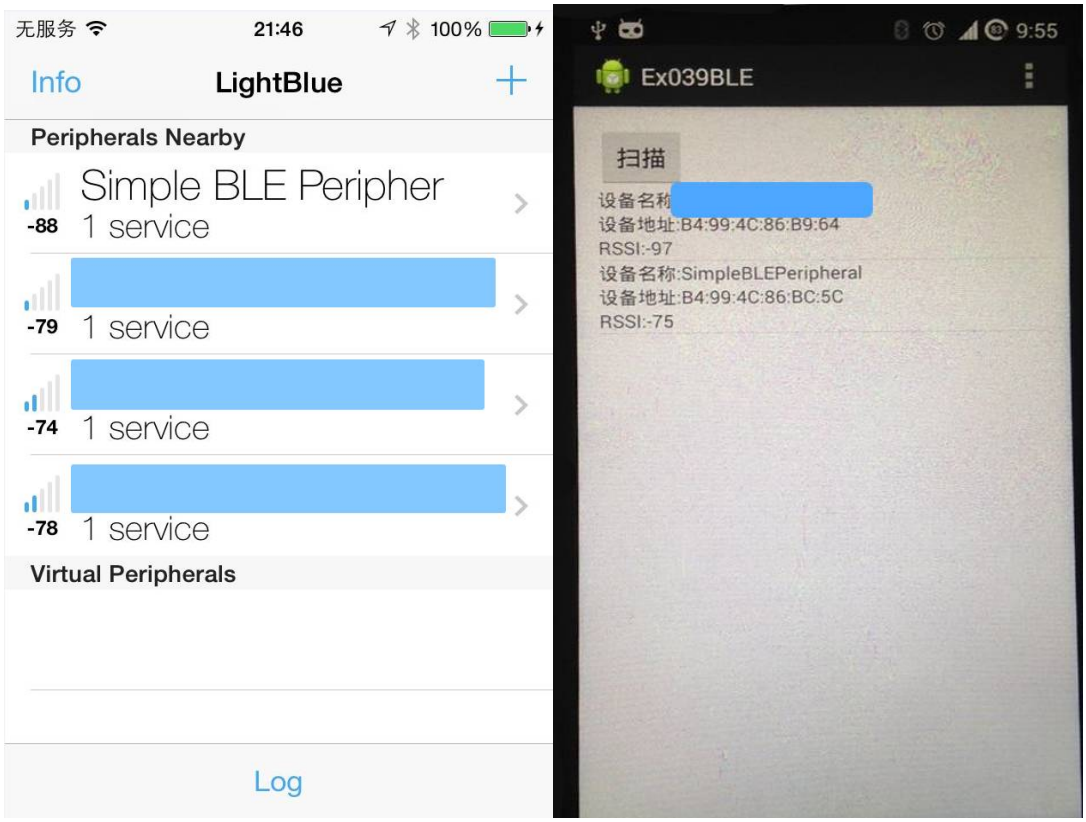
5.Click on the icon to download the program debug button on the device and enter debug mode. Figure。



6.Click Debug Step Out button on the column, enter the full speed。



7.Then open the phone BLE software, IOS (IPhone4s and above) on the LightBlue, Android (Android4.3 and above) on the Ex039BLE, refresh the list of Bluetooth, you will see a file called SimpleBLEPeripheral of Bluetooth peripherals. Figure。

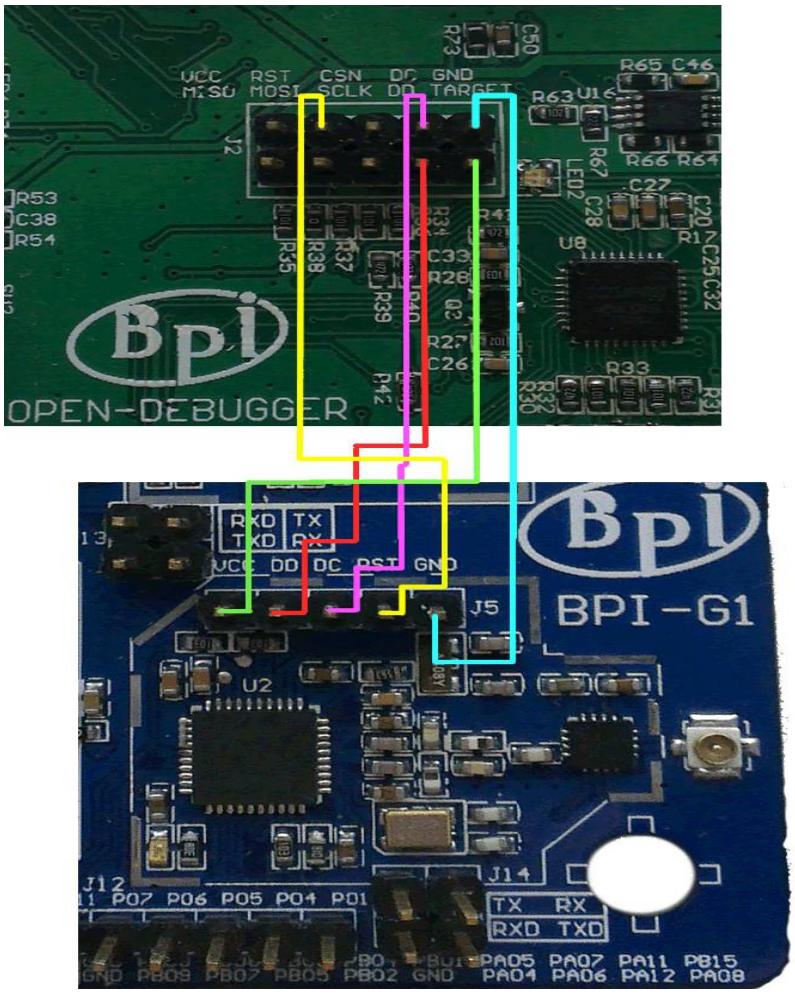


9.3 Quick start of TI CC2530

Download and install the following software：

- ZStack-CC2530-2.5.1a（Can go to google search to）
  - Assuming the default installation folder C:\Texas Instruments\ZStack-CC2530-2.5.1a
  - After installation is complete, go https://github.com/mark-legend/bpi_g1_zigbee_patch download hal_lcd.c file and overwrite the folder C:\Texas Instruments\ZStack-CC2530-2.5.1a\Components\hal\target\CC2530EB file with the same name
- IAR EW8051（The software is compiled BLE's IDE is the same, if you have installed, then the direct use, otherwise the installation yourself, we recommend using IAR EW8051 8.1）

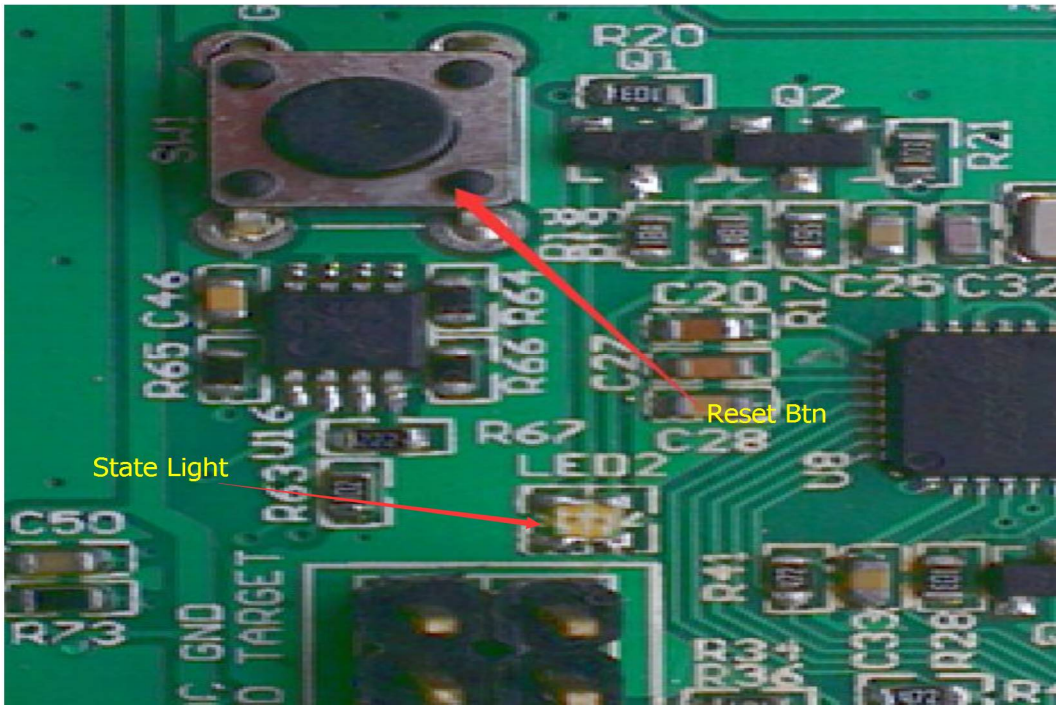
9.3.1 G1 connection BPI OPENDEBUGGER



9.3.2 BPI OPENDEBBUG check whether the software and hardware are connected

BPI OPENDEBUGGER connected to the PC, and check whether the drive is installed, if you can not identify, install SmartRF Studio， The software that comes with the driver. If you can identify that, please give G1 powered。

If the drive is normal, then check the G1 and OPENDEBBUG connection is normal. After the G1 to power, press the CC Debugger interface OPENDEBBUG reset button next。



If the connection is normal, then the status light next to the CC Debugger interface will turn green, if it is red or other colors, please check whether it is normal or electricity on the G1 G1 is properly connected and OpenDebugger。

9.3.3 Modifications to adapt the code for G1

1.Open the file C:\Texas

instruments\ZStack-CC2530-2.5.1a\Components\hal\target\CC2530EB\hal_board_cfg.h, jump to about 61 lines, will be as

```
/* ------------------------------------------------------------------------------------------------
 *                                          CC2590/CC2591 support
 *
 *                            Define HAL_PA_LNA_CC2590 if CC2530+CC2590EM is used
 *                            Define HAL_PA_LNA if CC2530+CC2591EM is used
 *                            Note that only one of them can be defined
 * ------------------------------------------------------------------------------------------------
 */
#define xHAL_PA_LNA
#define xHAL_PA_LNA_CC2590

```

Modified to

```
/* ------------------------------------------------------------------------------------------------
 *                                          CC2590/CC2591 support
 *
 *                            Define HAL_PA_LNA_CC2590 if CC2530+CC2590EM is used
 *                            Define HAL_PA_LNA if CC2530+CC2591EM is used
 *                            Note that only one of them can be defined
 * ------------------------------------------------------------------------------------------------
 */
#define HAL_PA_LNA
#define xHAL_PA_LNA_CC2590
```

2.Open the file C:\Texas Instruments\ZStack-CC2530-2.5.1a\Components\mac\low_level\srf04\single_chip\mac_radio_defs.c，jump to about 300 lines, will be as

```
{ /* either if compound statement or non-conditional compound statement */

  /* (Re-)Configure PA and LNA control signals to RF frontend chips.
  * Note that The register values are not retained during sleep.
  */

  /* P1_1 -> PAEN */
  RFC_OBS_CTRL0 = RFC_OBS_CTRL_PA_PD_INV;
  OBSSEL1       = OBSSEL_OBS_CTRL0;

  /* P1_4 -> EN (LNA control) */
  RFC_OBS_CTRL1 = RFC_OBS_CTRL_LNAMIX_PD_INV;
  OBSSEL4       = OBSSEL_OBS_CTRL1;

  /* For any RX, change CCA settings for CC2591 compression workaround.
   * This will override LNA control if CC2591_COMPRESSION_WORKAROUND
   * flag is defined.
   */
  COMPRESSION_WORKAROUND_ON();
```

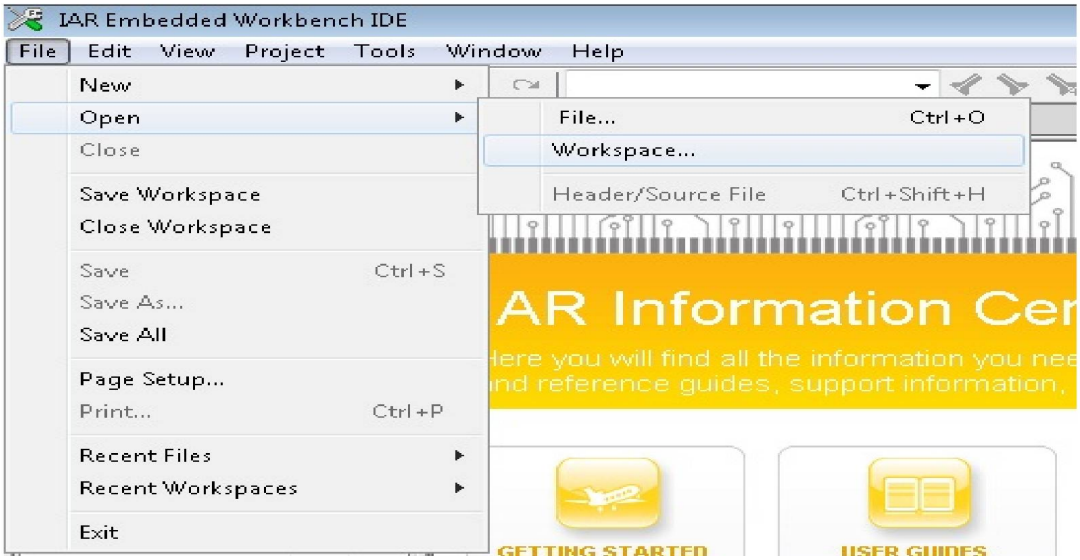
Modified to

```
{ /* either if compound statement or non-conditional compound statement */

  /* (Re-)Configure PA and LNA control signals to RF frontend chips.
  * Note that The register values are not retained during sleep.
  */

  /* P1_5 -> PAEN */
  RFC_OBS_CTRL0 = RFC_OBS_CTRL_PA_PD_INV;
  OBSSEL5       = OBSSEL_OBS_CTRL0;

  P1SEL &= ~0X10;
  P1DIR |= 0X10;
  P1_4   = 0x1;

  /* For any RX, change CCA settings for CC2591 compression workaround.
   * This will override LNA control if CC2591_COMPRESSION_WORKAROUND
   * flag is defined.
   */
  COMPRESSION_WORKAROUND_ON();
```

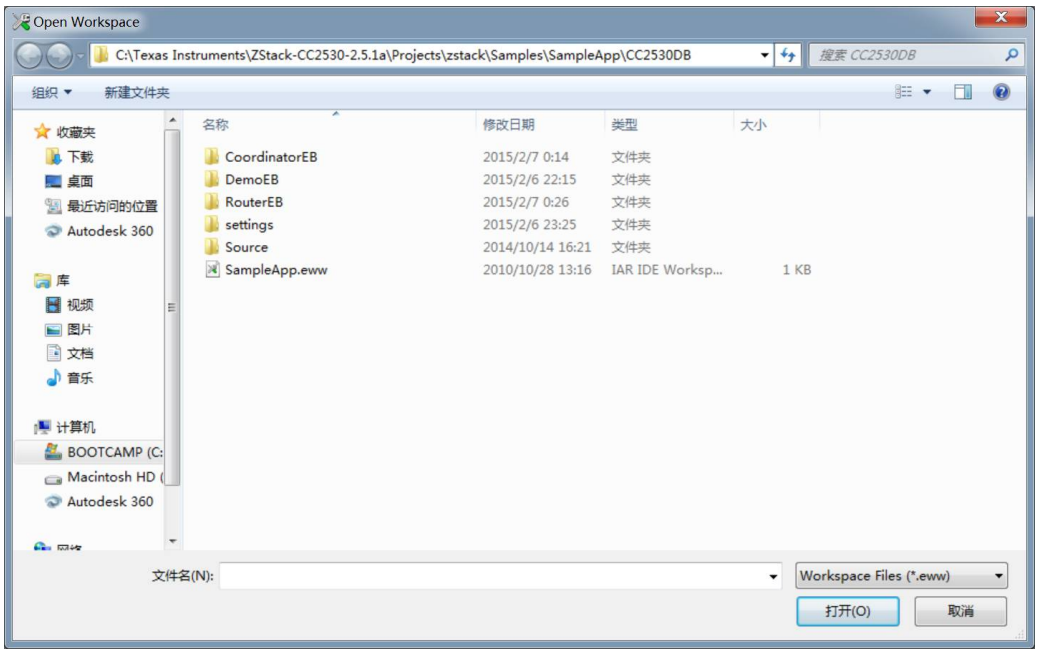
9.3.4 Compile, download and debug routines SampleApp

The operation requires two BPI G1 development board, and you need two OLED screens.

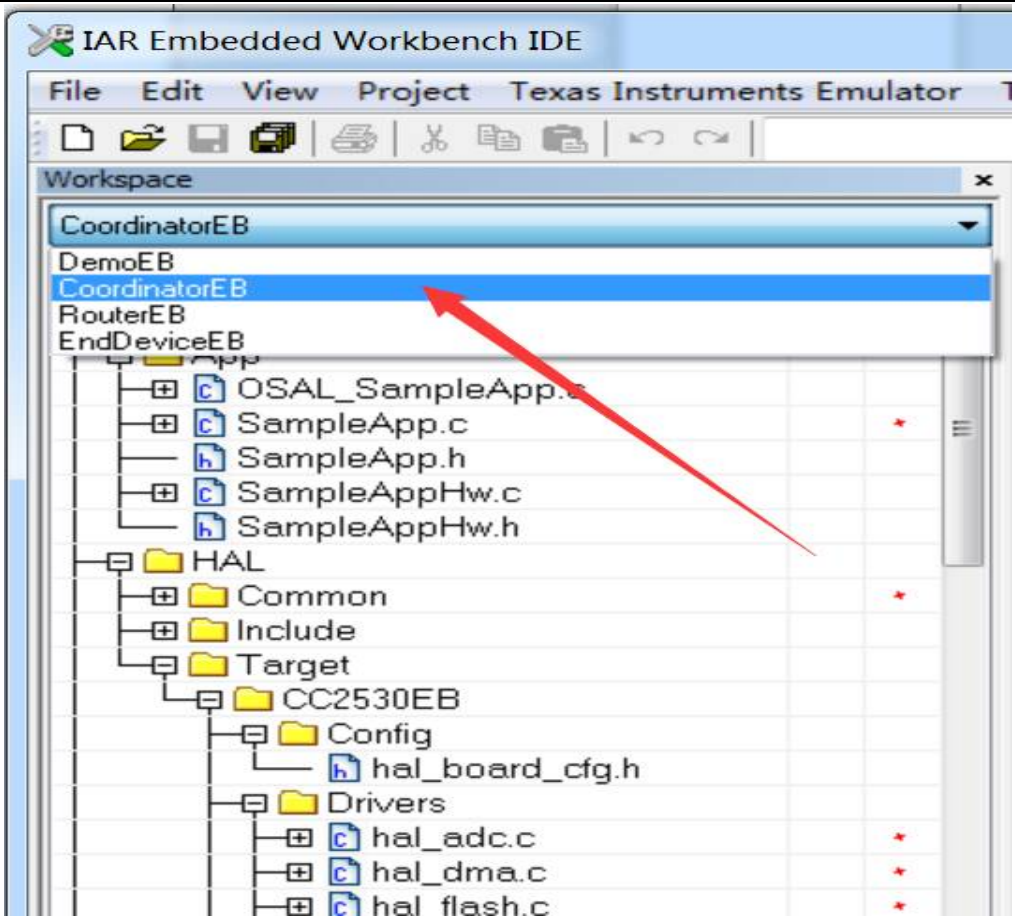
1.Start IAR EW8051 and open SampleApp project, select File> Open from the menu> Workspace。



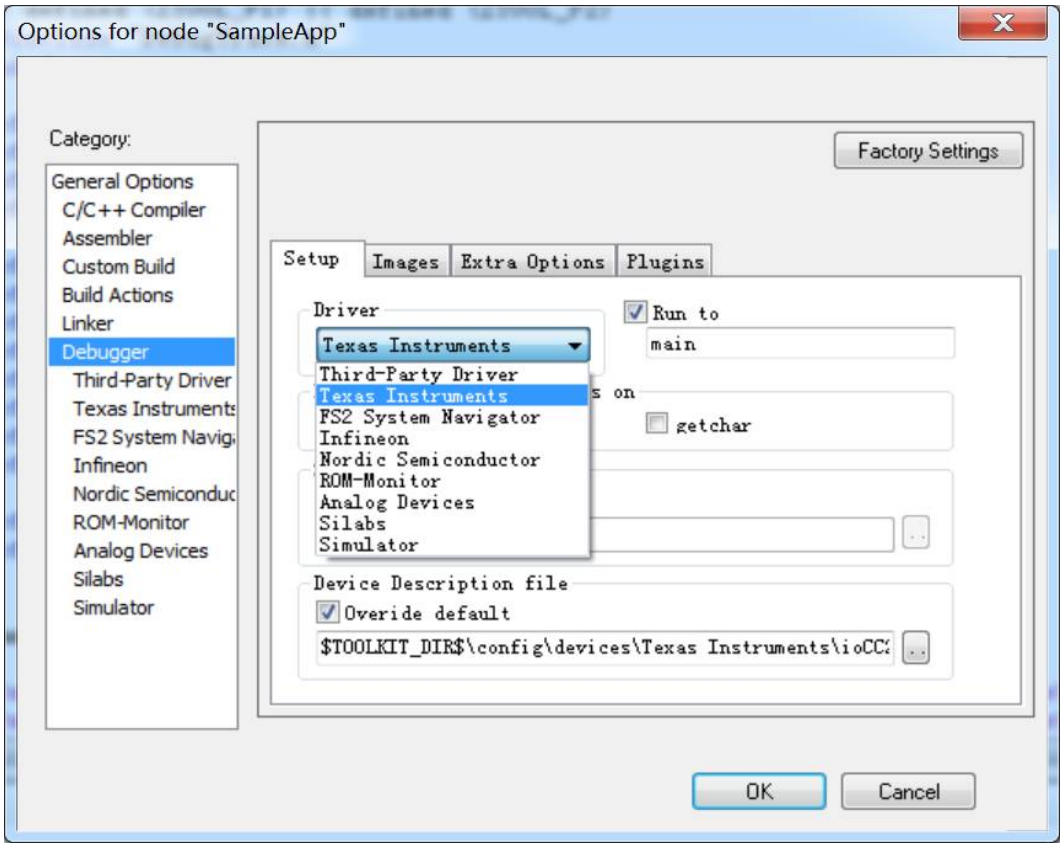
2.Browse to C:\Texas Instruments\ZStack-CC2530-2.5.1a\Projects\zstack\Samples\SampleApp\CC2530DB and open SampleApp.eww。



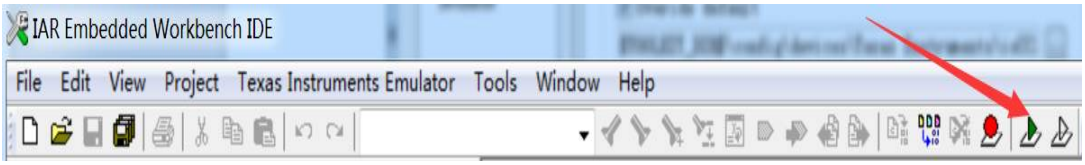
3.Choose from the list of items CoordinatorEB。



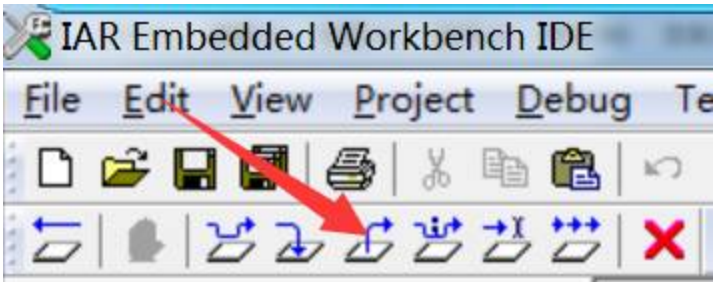
4.When debugging code must be downloaded to the device. Select Project> Options from the menu and select Debugger category. In this setup tab, select Texas Instruments as a driver, and then click OK. Figure。



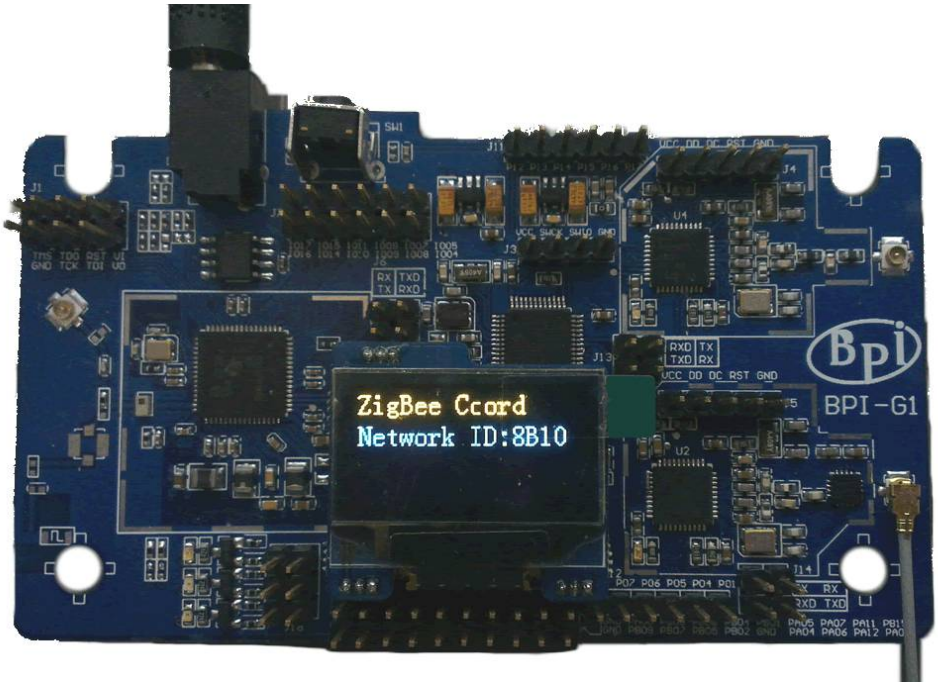
5.Click on the icon to download the program debug button on the device and enter debug mode. Figure。



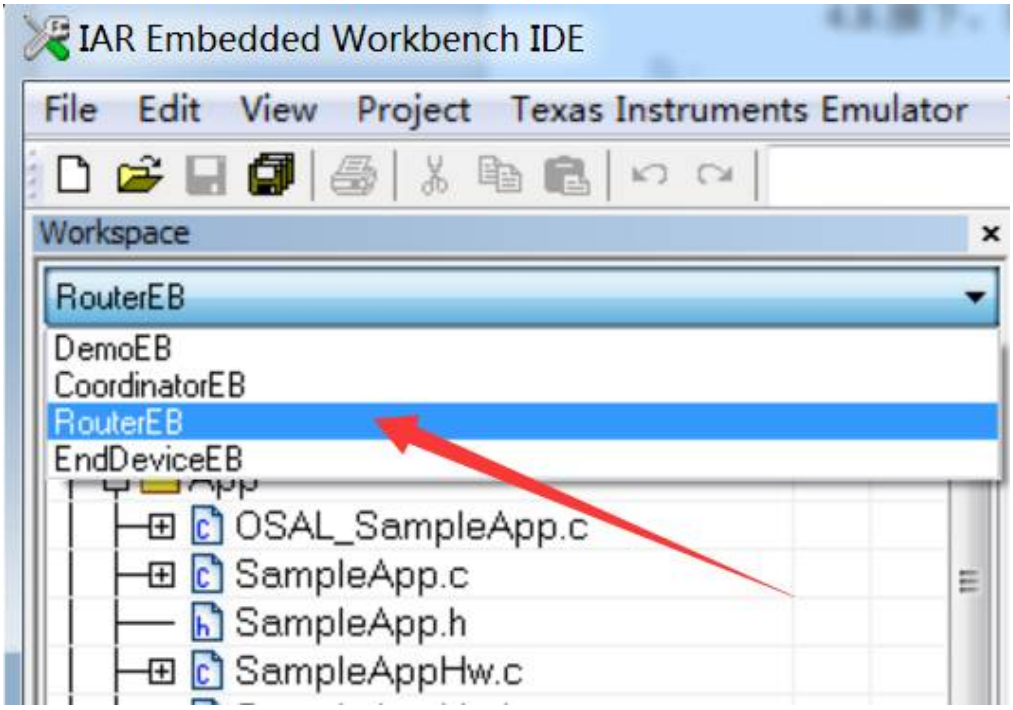
6.Click Debug Step Out button on the column, enter the full speed。



7.If there is no problem, a G1 should be is a good writer. Up and running in the following figure。

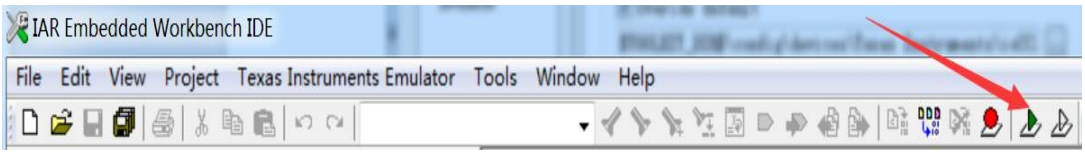


8.Next, another a G1 programming development board. Choose from the list of items RouterEB。

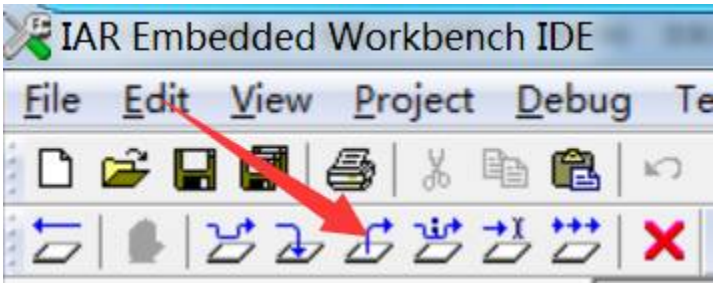


9.Another is a G1 and BPI OPENDEBUGGER connection. After connecting, click the CC Debugger interface again next reset button, look at the status light is green, if not green, check the power supply G1 or G1 with OPEN DEBUGGER the cable is properly。

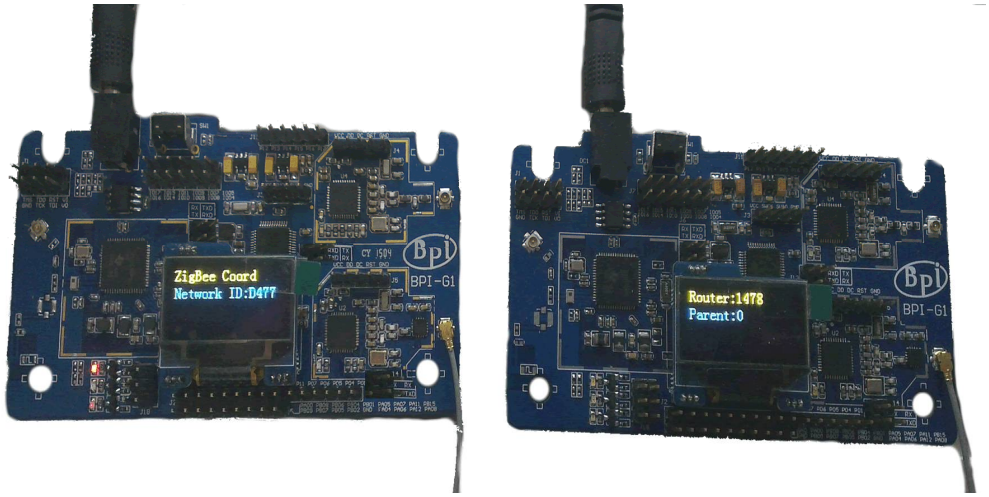
10.Click on the icon to download the program debug button on the device and enter debug mode. Figure。



11.Click Debug Step Out button on the column, enter the full speed。



12.If everything is normal, then the two G1 already up and running, you can see from the two G1 OLED screen they have connected to the。



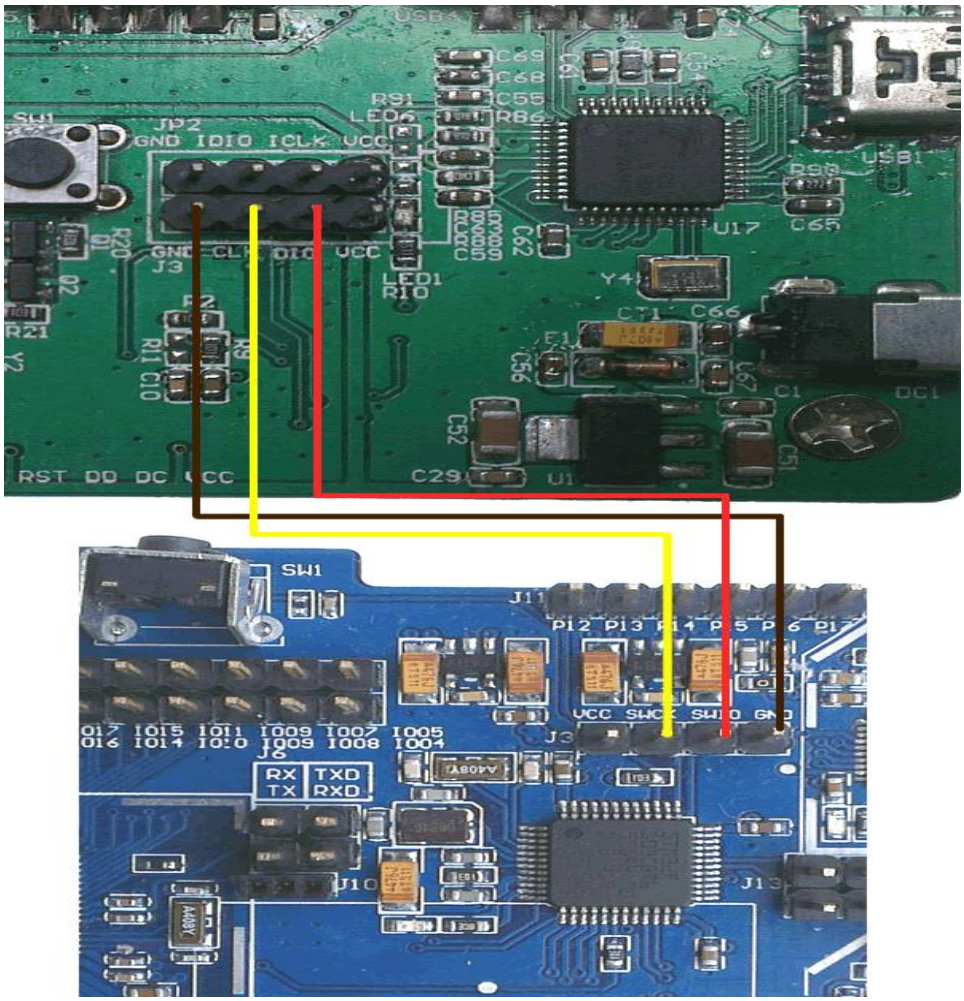
9.4 Quick Start STM32F103CB

Download and install the following software：

• IAR EWARM (the software with the IDE compiler CC3200 is the same, if already installed, then the direct use, otherwise the installation yourself, we recommend using IAR EWARM 7.2 or later)

• Go https://github.com/mark-legend/bpi_g1_stm32_simpleled download SimpleLed

9.4.1 G1 connection BPI OPENDEBUGGER

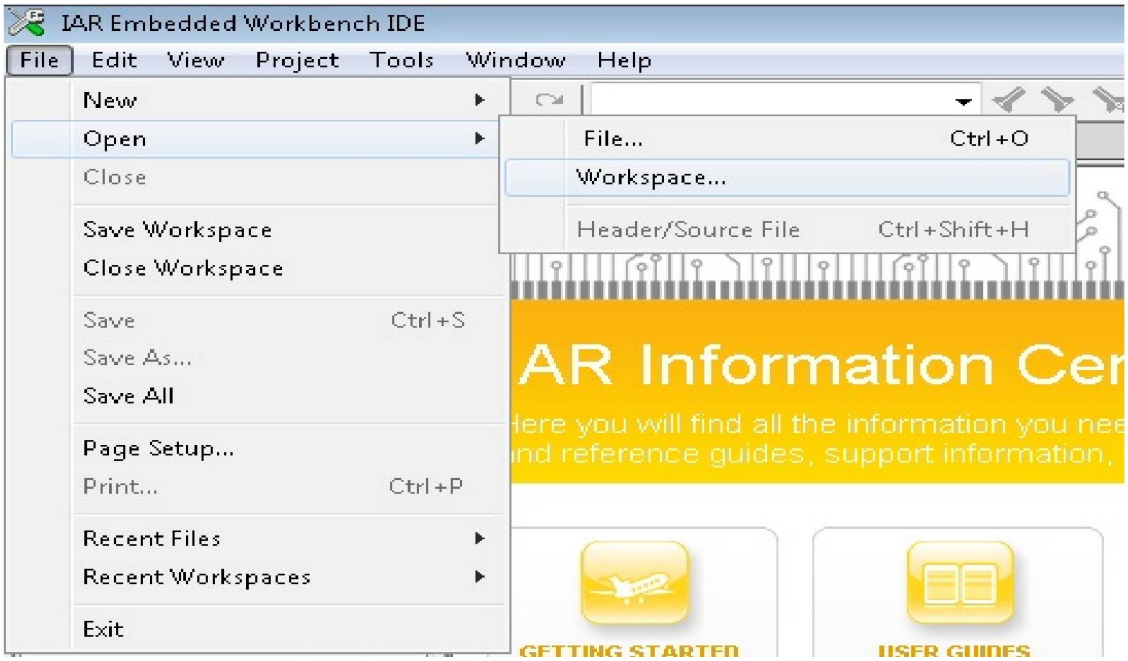


9.4.2 BPI OPENDEBBUG check whether the software and hardware are connected

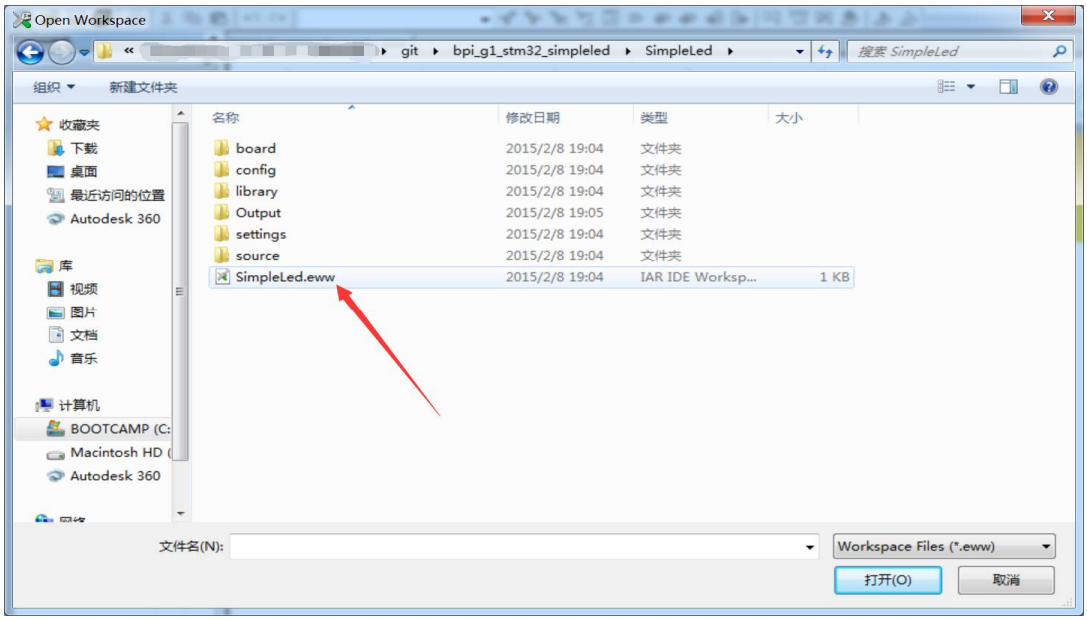
BPI OPENDEBUGGER connected to the PC, and check whether the drive is installed, if you can not identify, install JLinkARM，which comes with software drivers. If you can identify that, please give G1 powered。

9.4.3 Compile, download and debug routines SimpleLed

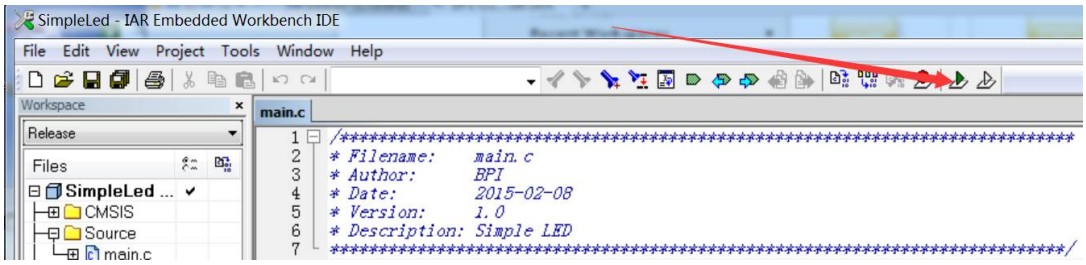
4.Start IAR EWARM and open SimpleLed project, select File> Open from the menu> Workspace。



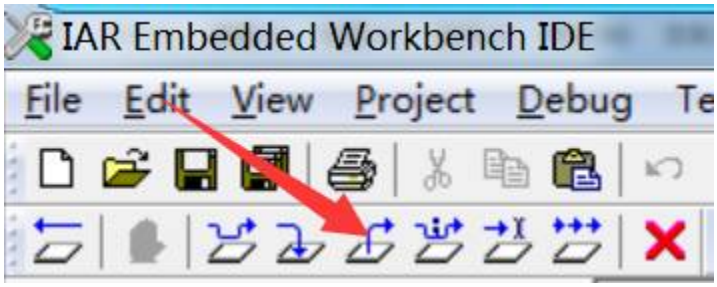
5.Browse to bpi_g1_stm32_simpleled \ SimpleLed and open SimpleLed.eww。



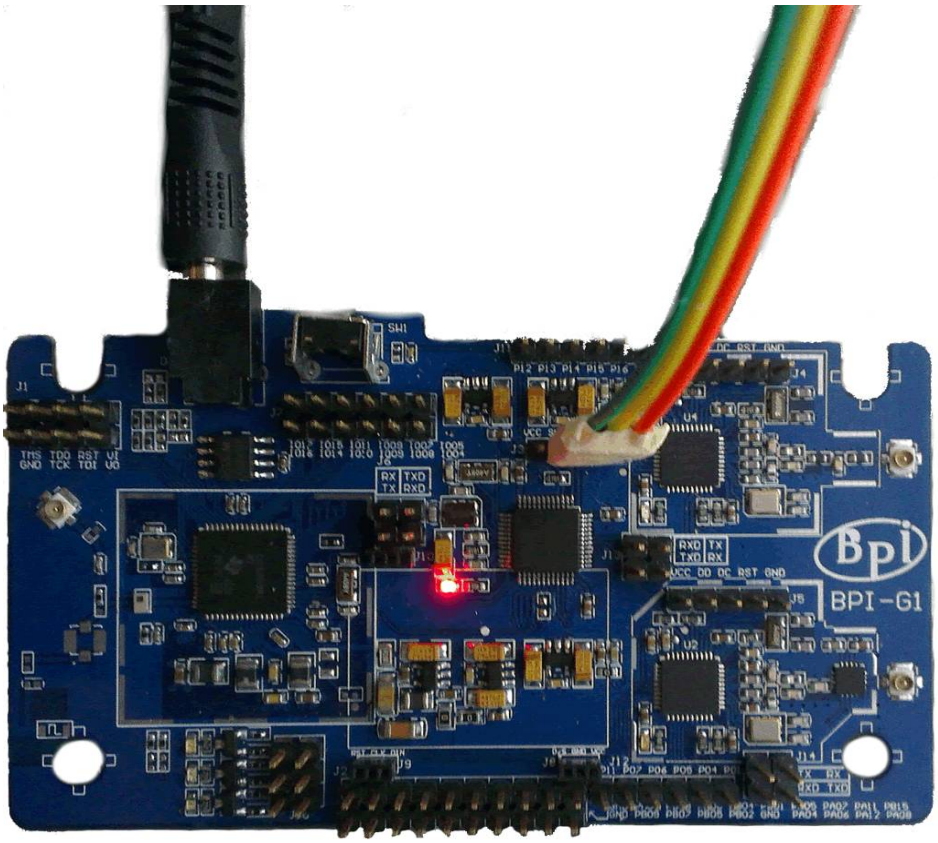
6.Click on the icon to download the program debug button on the device and enter debug mode. Figure。



6.Click Debug Step Out button on the column, enter the full speed。



8.If there is no problem, STM32 lower left corner of the LED will begin to flash lights。



10 Explanation

- In the implementation of the development of wireless applications on G1 plug in antenna。
- DC power input is 5V, the development board interfaces are 3.3V VCC。
- CC3200 The three status lights above the IO IO port and port are the same。